7-22-2010

Relating Multimodal Imagery Data in 3D

Karl C. Walli

Relating Multimodal Imagery Data in 3D

By

Karl C. Walli

B.S., Michigan Technological University, 1991

M.S., Joint Military Intelligence College, 1995

M.S., Rochester Institute of Technology, 2003

A dissertation submitted in partial fulfillment of the

requirements for the degree of Doctor of Philosophy.

Chester F. Carlson Center for Imaging Science

College of Science

Rochester Institute of Technology

July 22, 2010

Signature of the Author _____

Accepted by _____

Coordinator, Ph.D. Degree Program Date

CHESTER F. CARLSON CENTER FOR IMAGING SCIENCE

ROCHESTER INSTITUTE OF TECHNOLOGY

ROCHESTER, NEW YORK

CERTIFICATE OF APPROVAL

Ph.D. DEGREE DISSERTATION

The Ph.D. Degree Dissertation of Karl C. Walli has been examined and approved

by the dissertation committee as satisfactory for the dissertation

required for the Ph.D. Degree in Imaging Science

Dissertation Advisor: _____

Dr. John Schott

Committee Member: _____

Dr. Harvey Rhody

Committee Member: _____

Dr. Carl Salvaggio

External Chair: _____

Dr. Andrew Herbert

DISSERTATION RELEASE PERMISSION
CHESTER F. CARLSON CENTER FOR IMAGING SCIENCE
ROCHESTER INSTITUTE OF TECHNOLOGY

Title of Dissertation:

Relating Multimodal Imagery Data in 3D

I, Karl C. Walli, hereby grant permission to Wallace Memorial Library of R.I.T. to reproduce my dissertation in whole or in part. Any reproduction will not be for commercial use or profit.

Signature _____

Date

Disclaimer

The views expressed in this dissertation are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

Relating Multimodal Imagery Data in 3D

By

Karl C. Walli

Chester F. Carlson Center for Imaging Science

College of Science

Rochester Institute of Technology

ABSTRACT

This research develops and improves the fundamental mathematical approaches and techniques required to relate imagery and imagery derived multimodal products in 3D. Image registration, in a 2D sense, will always be limited by the 3D effects of viewing geometry on the target. Therefore, effects such as occlusion, parallax, shadowing, and terrain/building elevation can often be mitigated with even a modest amounts of 3D target modeling. Additionally, the imaged scene may appear radically different based on the sensed modality of interest; this is evident from the differences in visible, infrared, polarimetric, and radar imagery of the same site.

This thesis develops a 'model-centric' approach to relating multimodal imagery in a 3D environment. By correctly modeling a site of interest, both geometrically and physically, it is possible to remove/mitigate some of the most difficult challenges associated with multimodal image registration. In order to accomplish this feat, the mathematical framework necessary to relate imagery to geometric models is thoroughly examined. Since geometric models may need

to be generated to apply this 'model-centric' approach, this research develops methods to derive 3D models from imagery and LIDAR data. Of critical note, is the implementation of complimentary techniques for relating multimodal imagery that utilize the geometric model in concert with physics based modeling to simulate scene appearance under diverse imaging scenarios. Finally, the often neglected final phase of mapping localized image registration results back to the world coordinate system model for final data archival are addressed.

In short, once a target site is properly modeled, both geometrically and physically, it is possible to orient the 3D model to the same viewing perspective as a captured image to enable proper registration. If done accurately, the synthetic model's physical appearance can simulate the imaged modality of interest while simultaneously removing the 3-D ambiguity between the model and the captured image. Once registered, the captured image can then be archived as a texture map on the geometric site model. In this way, the 3D information that was lost when the image was acquired can be regained and properly related with other datasets for data fusion and analysis.

Table of Contents

Page

List of Figures

Page

Glossary

Bundle Adjustment. A photogrammetric process utilized to relate multiple cameras, images and the resulting sparse structure by solving for the camera's external and internal parameters w.r.t. corresponding image control points.

Dense Point Cloud (DPC). An array of 3D points, that is often associated with a LIDAR dataset and is described by a global coordinate system.

Discrete Linear Transform (DLT). A linear technique that can provide an initial estimate of a solution space that is often desired to seed a non-linear optimization algorithm.

Exterior Orientation Parameters (EOP). These parameters refer to the location of the camera lens [X, Y, Z] and the orientation of the camera [ω, ϕ, κ] at the time of image capture.

Faceted Models (FM). This refers to the traditional computer graphics models that contain vertices and facets to represent the 3D structure of a scene.

Interior Orientation Parameters (IOP). These parameters refer to the intrinsic properties of the camera and include focal length, principle point, focal plane skew, and radial distortion.

Levenberg-Marquardt Algorithm (LMA). A robust nonlinear optimization technique often used in computer vision problems for estimating the solution to nonlinear least squares problems.

Random Sample Consensus (RANSAC). A technique for robustly removing outliers from a dataset. It does this by minimally sampling the data a statistically significant number of times to create a mathematical model that maximizes the number of inliers within an error region.

Space Resectioning. A photogrammetry term that implies solving for a camera's pose by relating points in one image to those in another, or to a model.

Sparse Bundle Adjustment (SBA). The term "sparse" here relates to the sparse matrix techniques utilized to solve for extremely large, but, weakly correlated parameters involved when solving for most Bundle Adjustments.

Sparse Point Cloud (SPC). The array of 3D points locally defined within a 3D coordinate system.

Sparse Structure Bundle (SSB). This includes the entire bundle of sparse structure, images and related camera positions within a common and local 3D coordinate system.

UV Texture Map. A standard technique in the graphic modeling community used to realistically texture 3D models. This technique maps a composite texture, mapped in the normalized '$uv\ plane$', to the vertices of select model facets; thus obtaining the name "UV Texture Map".

World Coordinate System (WCS). The absolute coordinate system linked to the global grid.

1 Introduction

1.1 Use of Imagery Data is now Mainstream

Over the course of the last decade, the use of imagery based products from airborne and satellite platforms have become mainstream. Applications like Google Earth/Maps (Google Earth 2010) and Bing Maps (Microsoft Corporation 2010) allows a user to plan travel, assess real-estate, teach their children geography, or visualize where the latest 'crisis du jour' is happening in the world at the click of a mouse button. The ability to seamlessly view hundreds of integrated image products in visual databases has thrown open the doors on the once "niche field" of imagery analysis, integration, fusion, and database archival.

The VanLare Water Processing Plant – Google Earth Software View

Figure 1-1 – This Google Earth (Google Earth 2010) view of the VanLare Site contains a hi-fidelity model courtesy Pictometry Int. (Pictometry 2010) and is representative of the realistic representations possible within today's GIS environments.

Along with this keen new interest by the general population in seeing a "bird's eye view" of the world, comes new mathematical advancements from the field of computer vision that are allowing robots to perceive their surroundings and avoid obstacles. What do these two observations have in common? They both require the processing of large volumes of imagery that are captured from a multitude of vantage points, registered together, and provided in local or global 3D coordinate systems that allow for integration, fusion and archival.

1.2 The Problem

Although great strides have been made in the automated registration of grayscale imagery from similar viewing geometries, there are still great challenges in developing robust automated techniques for registering images taken from varying viewing geometries and from different spectral modalities. The challenges for 3D multimodal registration are many and are directly linked to the angular and spectral disparity of the datasets themselves (Van Nevel 2001). The 3D influences of the scene-to-sensor viewing geometry creates occlusions and parallax effects, the changing solar illumination causes varying shadow positions, and the diverse appearance of the scene due to a sensor's spectral responsivity ensures the continuing difficulty in automatically registering and relating remotely sensed imagery of a site (Figure 1-2).

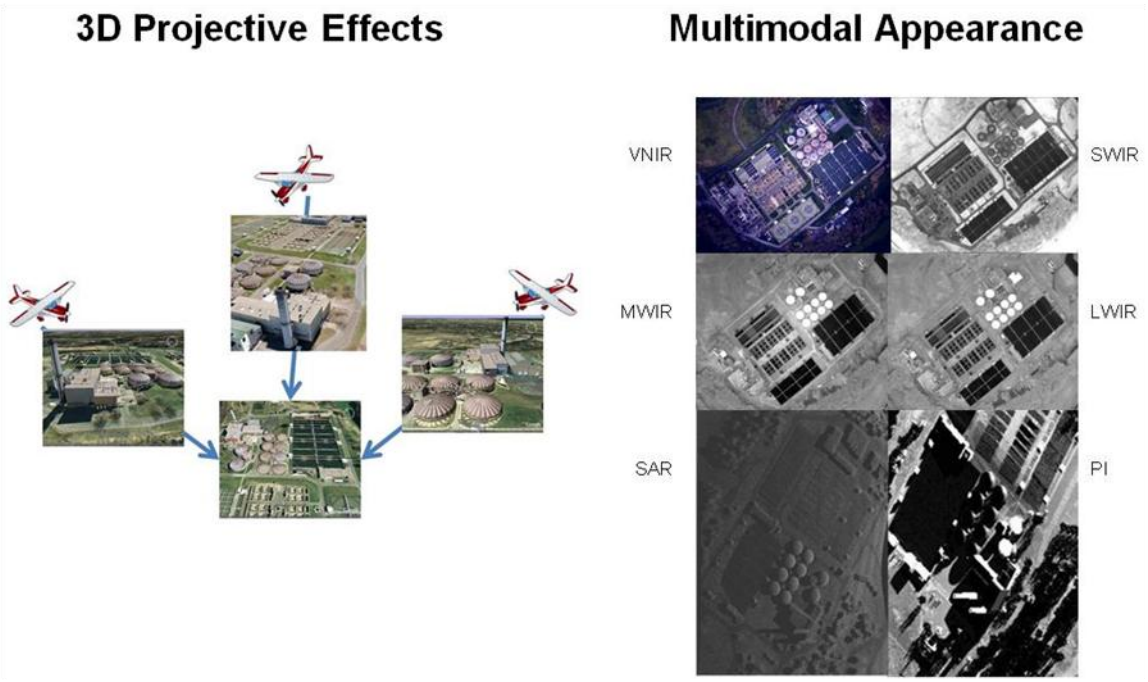


Figure 1-2 The registration challenges resulting from viewing geometry including parallax, occlusion, & shadowing (left) and spectral diversity (right) are visible above (synthetic SAR and PI images of VanLare courtesy Dr Mike Gartley).

1.3 The Solution

For decades, imagery analysts (the author included) have tried to register images within a 2D construct only to find that this solution space is barely adequate to accomplish the task at hand. It should always be kept in mind that an image is a projection of the 3D world from a certain vantage point. This 2D projection contains all of the 3D influences of the environment including the terrain, foliage and the buildings. A 2D solution to relating imagery is only justified when these images are taken from similar vantage points or if the 3D influences are negligible, such as when the terrain is flat or if these influences have been removed through ortho-rectification. It should be no surprise to those that have been frustrated with the limitations of 2D image registration, that this 3D problem necessitates a 3D solution.

In previous work by the author (Walli, Multisensor Image Registration utilizing the LoG Filter and FWT 2003), a case study was developed that demonstrated the results of an automated 2D image registration algorithm over an urban section of San Diego, CA that contained large amounts of terrain relief and building parallax (Figure 1-3). These images were taken with enough angular disparity to exhibit significant amount of parallax, thus frustrating automated registration attempts with low error.

The Limitation of 2D Image Registration

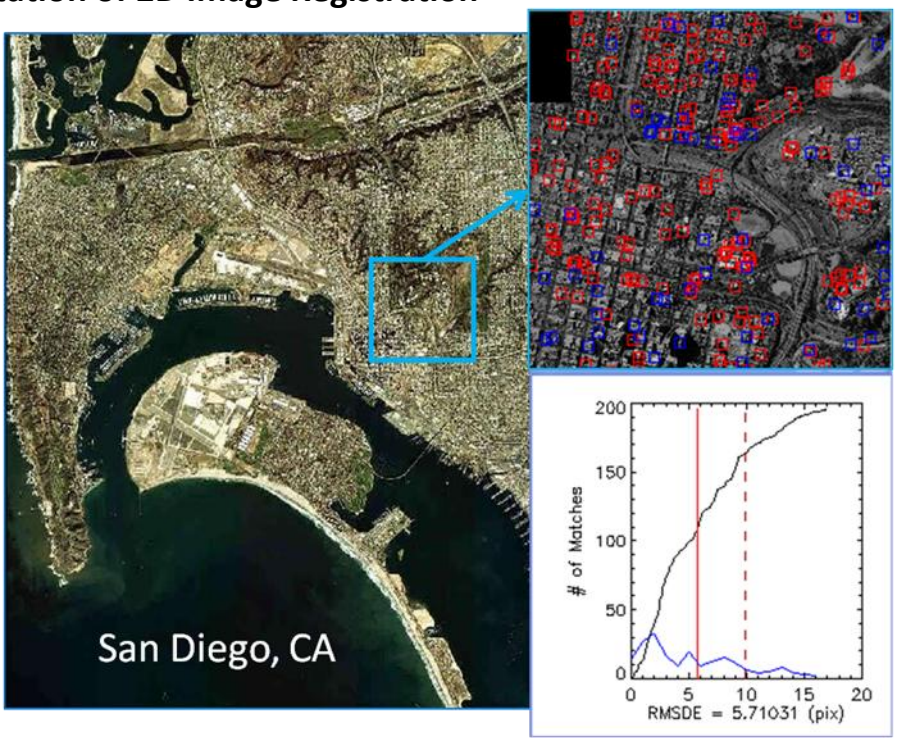


Figure 1-3 This graphic shows the result of registering two images of San Diego, where ~75% of the correctly matched features (red squares) were discarded in a vain attempt to obtain subpixel registration accuracy to a 2D model.

In this example, the 1 meter resolution Ikonos imagery (GeoEye, Inc 2010) was used to obtain ~200 good feature matches. Unfortunately, these correspondences resulted in a rather poor error analysis result, when attempting to relate them using a 2D transformation. Even after considerable refinement/culling of ~75% of the matched feature locations, through error

analysis and removal (covered in Section 2.5), the final registration provided only mediocre results. This is because the 2D solution space was inadequate in its dimensionality to encompass the matched features, which were highly nonplanar. This dilemma provided a great deal of justification for the author to pursue a full 3D solution to the image registration problem, especially as it pertained to the challenges of accurately fusing multimodal imagery data for the project described below.

1.4 The Advanced ANalyst Exploitation Environment (AANEE)

The AANEE program was conceived by Dr John Schott as a demonstration of what could be accomplished if the current "state-of-the-art" in synthetic scene modeling, image registration, and process modeling were combined in a seamless virtual environment for an intelligence analyst. The main thrust of this project is to immerse an analyst within an environment where the datasets are archived in a visual database that is easy to interact with and where the data can be interrogated in an intuitive fashion.

In the world of AANEE, a user could fly through a scene, stop at a building of interest and click on a wall. Once this is done, the building wall would verbally tell the user when it was made along with other historical facts. The user would then have pull-down menu options that would allow for temporal playback of imagery that might highlight any change to the building over time. Additionally, the user may request imagery that has been collected in multi-modal spectra other than the traditional visual RGB or Panchromatic bands shown in (Figure 1-4).

The VanLare Water Processing Plant – AANEE Software View

Figure 1-4 – The scenes above show the same model of the VanLare Plant from within the AANEE software program. Note the accurate casting of shadows and the ability to predict occlusions due to the 3D modeled site and landscape.

To enable AANEE to be more than just a game simulation environment, it is necessary to be able to use the 3D scene model as a "skeleton" from which to project layers of imagery products for immediate visual inspection and long term archival. Because once an imagery based product is registered to an accurate 3D model, it is possible to regain the 3D nature of the scene that was lost when the image was acquired, but only if it is projected back onto the model from the same vantage point that it was taken. In this manner, a 3D database of archived imagery can be saved as image textures on the model and can be categorized temporally, spectrally, and of course spatially as seen in Figure 1-5.

The VanLare Water Processing Plant – AANEE Software View

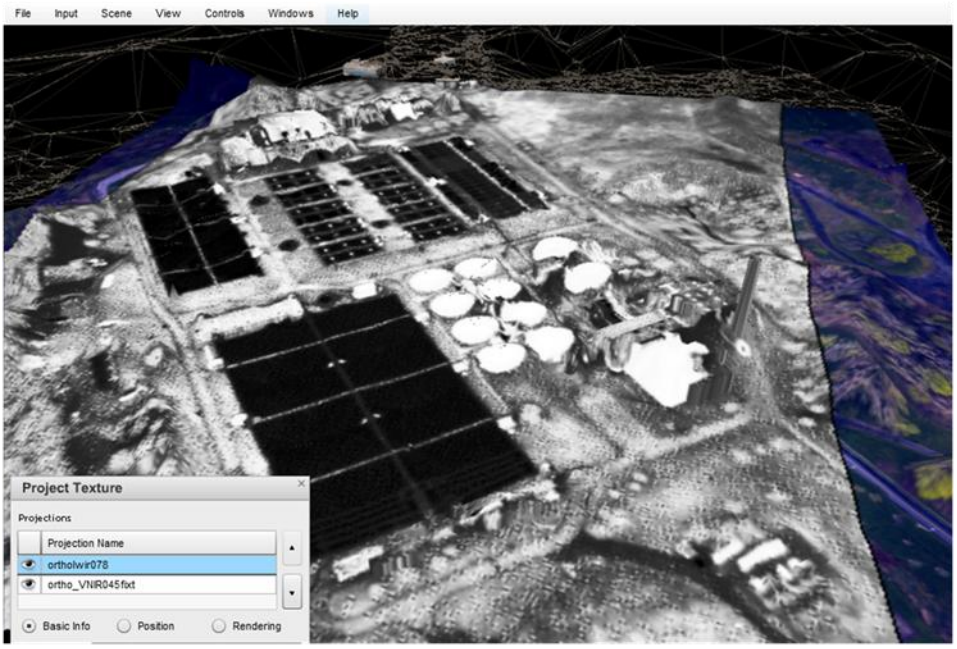


Figure 1-5 - This view of the VanLare site from the AANEE software program contains projections of additional RGB and LWIR data from RIT's WASP sensor over the site and terrain model, before registration; using only sensor IMU/GPS data.

1.5 The 3D Model as an Archival Database

Recent advancements in computer vision (epipolar geometry) provide the ability to understand and model our world in 3D. This allows elegant new solutions to tough old image registration problems such as understanding and compensating for the effects of scene projection while relating common features from a database of images. Additionally, a hi-fidelity 3D model of a scene can help predict and mitigate the effects of occlusion and shadowing if the orientation of the model (pose) can be determined at the time of image acquisition.

Knowledge of these challenges are critical for understanding the author's 'model-centric' approach to registration and so a significant portion of this document will be spent in developing techniques (Chapters 2-5) that will be utilized to mitigate these effects. The need for a 3D Model, for accurate registration of most visible band imagery products, is augmented

by the need for a physical model when registration of multi-modal imagery is required. The author will show how physics based modeling of a scene using the Center for Imaging Sciences (CIS) Digital Imagery and Remote Sensing Image Generation (DIRSIG) program can be utilized to simulate multimodal imagery that is good enough to automatically register to real data (Section 6.3). This will allow for DIRSIG to act as a physical Rosetta Stone for relating a potentially large range of disparate imagery products. The 'model-centric' approach to relating data and how DIRSIG is utilized to enable multi-modal image registration is covered in detail in Chapter 6.

1.6 Summary

With the growing interest in integration and fusion of imagery based data, fundamental research is required in the vital area of mathematical data-relationship development and database archival. The author has been continually amazed at how often "well registered" data is taken for granted as an assumption in both fusion applications and change detection scenarios. Neglecting the essential step of developing a framework to properly relate the data in a true 3D sense is to ignore the sensor acquisition pose and the structure in a scene and the effect that they can have on the final registered product. Both image modality fusion and change detection algorithms should perform at their best when the initial data has been accurately related in 3D.

The research covered herein develops the fundamental mathematical approaches and techniques required to relate multimodal imagery and imagery derived products in 3D. Additionally, it improves upon some well established methods for relating imagery derived products, by applying new epipolar geometry and efficient mathematical techniques. Finally,

the author's physical modeling approach to relating multimodal imagery is a cornerstone of the value added research contained within this document. The figure below depicts the five major subcategories that will be covered in this research and their related sections in the document.

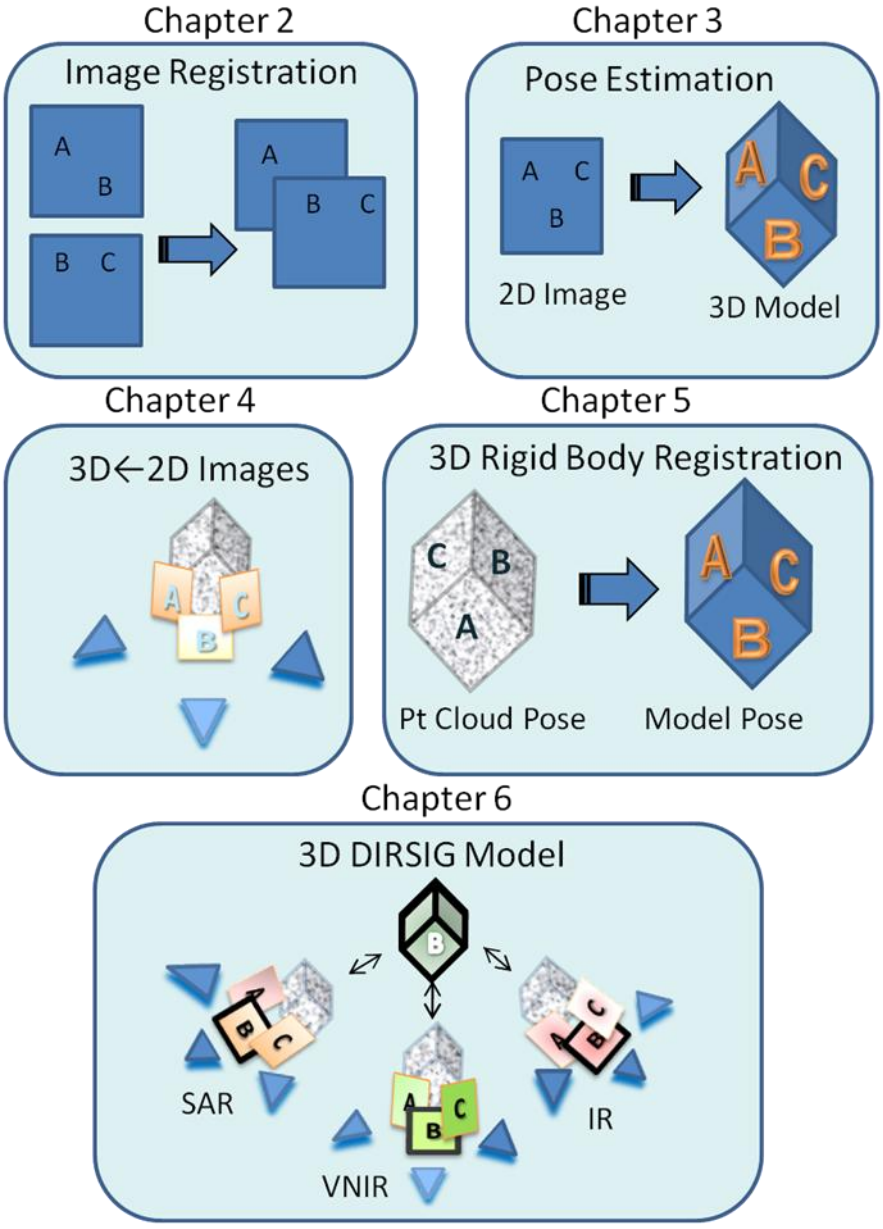


Figure 1-6 The five primary areas of research contained in this dissertation are covered in Chapters 2-6.

2 Image Registration

Image registration, in a 2D sense, will always be limited by the 3D effects of viewing geometry on the target. Therefore, effects such as occlusion, parallax, shadowing, and terrain/building elevation can often be mitigated with even a modest amounts of 3D target modeling. Once a target is modeled and textured with representative imagery, it is possible to orient the scene based model to the same viewing perspective as any remotely sensed image to enable proper registration. If done accurately, the 3-D ambiguity between the model and the image can be removed and the newly registered image can now be utilized as an additional texture layer on the model. If this is done with enough precision, the 3D information that was lost when the image was acquired can be regained and properly related to other imagery and data of the target scene. The basic process for registering two images is provided below in Figure 2-1.

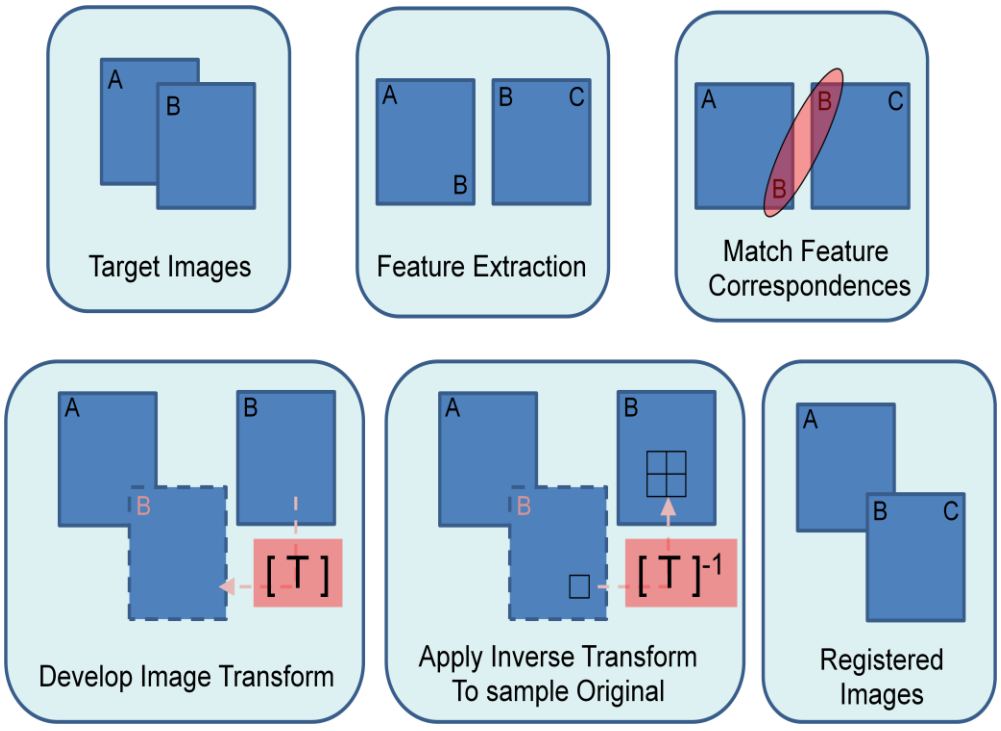


Figure 2-1 The basic process for automatically relating images.

2.1 Invariant Feature Extraction

Due to the significant amount of research into automated image registration over the years, there are several techniques that have been developed that work reasonably well. Currently, the most robust techniques appear to be multiscale edge based techniques due to their robust ability to extract repeatable structure from within a scene, even when relating multimodal imagery. For this reason, the choice of filters to help identify and extract these invariant edge features from images is a critical design decision for any automated registration process.

In detailed experimentation (K. Mikolajczyk 2002), it was found that the maxima and minima of a normalized version of the Laplacian of Gaussian (LoG) produce the most stable image features compared to a range of other possible image functions, including the gradient, Hessian, and Harris Corner Detector (Harris and Stephens 1988). Due to the proven performance of the LoG filter and its Difference of Gaussian (DoG) approximation, to robustly extract invariant features from imagery, these two filters will be explored further.

2.1.1 Laplacian of Gaussian (LoG) Filter

The idea for using edge detection filters for robust feature extraction was sparked while performing research into automated image registration (Walli, Multisensor Image Registration utilizing the LoG Filter and FWT 2003). It quickly became apparent that the LoG filter could be utilized to consistently pinpoint features within an overhead image that might be utilized for image registration. By applying a threshold to the LoG filtered image, it is possible to isolate regions that have similar rates-of-variation within a scene and to do so in a repeatable fashion. This is due to the "second derivative" (∇^2) nature of the Laplacian filter which produces high

output for well defined edges. Figure 2-2 demonstrates the effect of the LoG filter on a synthetic dataset that resembles the letter "X" but could represent a crossroads or building in an overhead image.

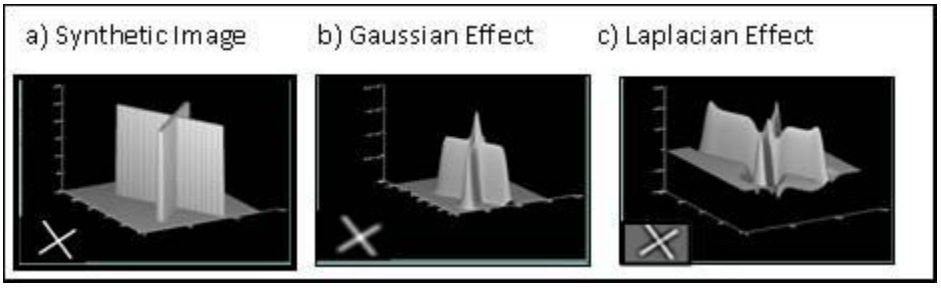


Figure 2-2 Demonstration of the LoG filter effects on synthetic edge data.

The effect that the LoG filter has on an image is very similar to the lateral-brightness adaptation of the human eye (also known as lateral inhibition) that leads to the "Mach band effect". Evidence of this is provided by Gonzalez and Woods, when they maintain that certain aspects of human vision can be modeled mathematically in the basic form of the LoG equation (Gonzalez and Woods 2007). This phenomenon is demonstrated in Figure 2-3, with an exaggeration of grayscale edge steps.

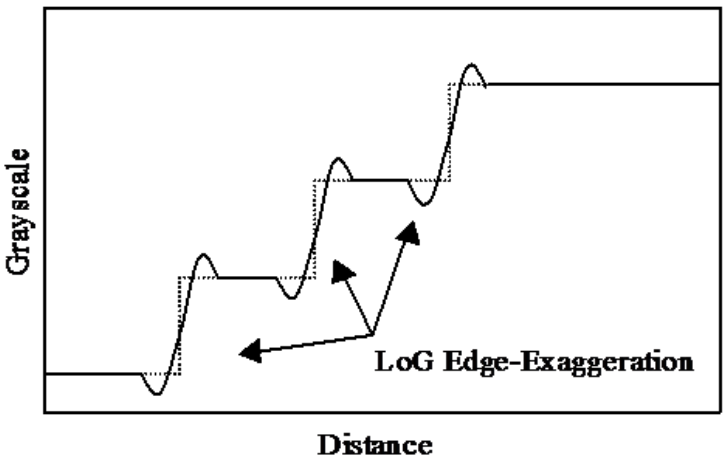


Figure 2-3 Edge-exaggeration resulting from convolution with a 1D LoG filter

The Laplacian is the second derivative of a function. This equation takes the following forms for both the 1-D and 2-D versions, as shown in (1) and (2):

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} \tag{1}$$

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \tag{2}$$

Additionally, this function can be approximated with the following 1-D & 2-D digital filters as seen below in (3) and (4):

$$\nabla^2 = [1 \quad -2 \quad 1] \tag{3}$$

$$\nabla^2 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix} \tag{4}$$

A graphical representation of the effects of this filter when applied to a 1-D step function (Figure 2-4.a) that has been first convolved with a Gaussian low-pass filter (Figure 2-4.b) follows. It can be seen why the 2nd Derivative filters are also called “zero-crossing” edge detectors since the knife edge input (Figure 2-4.a) goes to unity precisely at the zero crossing between the positive and negative peaks of Figure 2-4.d.

Although the LoG filter can be easily deconstructed into its component parts as seen below, it is more commonly implemented in one convolution step with a kernel similar to Figure 2-5.

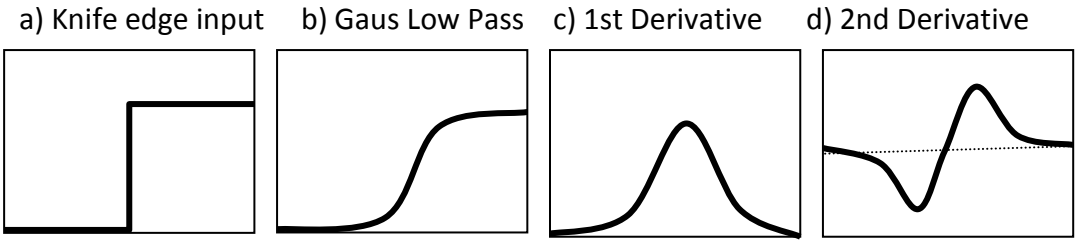
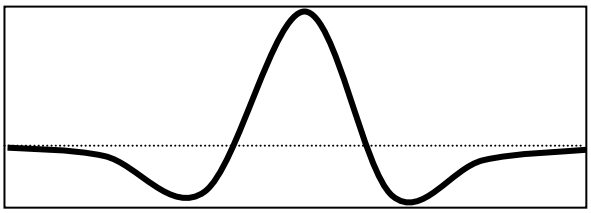


Figure 2-4 Visual effect of the Laplacian of Gaussian Filters in succession.

The 5x5 filter approximation and the "Mexican Hat" (LoG) function are shown below.



0	0	-1	0	0
0	-1	-2	-1	0
-1	-2	16	-2	-1
0	-1	-2	-1	0
0	0	-1	0	0

Figure 2-5 A 1-D representation of the LoG function and the composite 5x5 approximated filter.

The Laplacian is very good at highlighting variation within an image. This result is useful if the variation is equivalent to information content or edges, but, detrimental if that variation is represented by noise. On its own, the Laplacian will accentuate all high frequency components, including noise, along with the edges. For this reason the image is first convolved with a Gaussian filter, to diminish the effects of noise, before the Laplacian filter is applied.

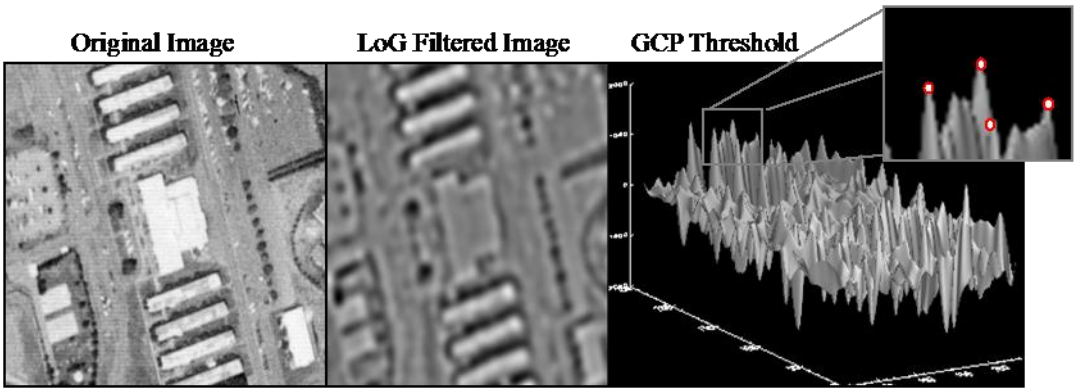


Figure 2-6 The results of the LoG filter and thresholding of maxima to create Control Points.

The results of the LoG thresholding process provide automated Ground Control Point (GCP) feature extraction within each image, as seen in Figure 2-6. Once these GCPs have been identified, a point matching routine (Section 2.2) can be utilized to relate the subset of similar points from each image. These related points can then be used to develop a transformation equation, for registration of the two images.

2.1.2 Difference of Gaussian Filter

The Difference of Gaussian (DoG) Filter is an approximation to the Laplacian of Gaussian Filter (Gonzalez and Woods 2007). Like the LoG, the image is first blurred with a low-pass Gaussian convolution filter which has an initial $width = \sigma_1$, where the Gaussian is mathematically described by,

Gaussian Function

$$G_{\sigma_1}(x,y) = \frac{1}{\sqrt{2\pi\sigma_1^2}} e^{\left(\frac{x^2+y^2}{2\sigma_1^2}\right)} \quad \textbf{(5)}$$

The image can be smoothed using two different Gaussian widths (σ_1 and σ_2) as shown below in Equations (6) and (7).

Image Blurred w/ σ_1

$$g_1(x,y) = G_{\sigma_1}(x,y) * f(x,y) \quad \textbf{(6)}$$

Image Blurred w/ σ_2

$$g_2(x,y) = G_{\sigma_2}(x,y) * f(x,y) \quad \textbf{(7)}$$

Now the Difference of Gaussian can be accomplished by subtracting the two blurred images ,

DoG Filter

$$g_1(x,y) - g_2(x,y) = (G_{\sigma_1} - G_{\sigma_2}) * f(x,y) = DoG * f(x,y) \quad \textbf{(8)}$$

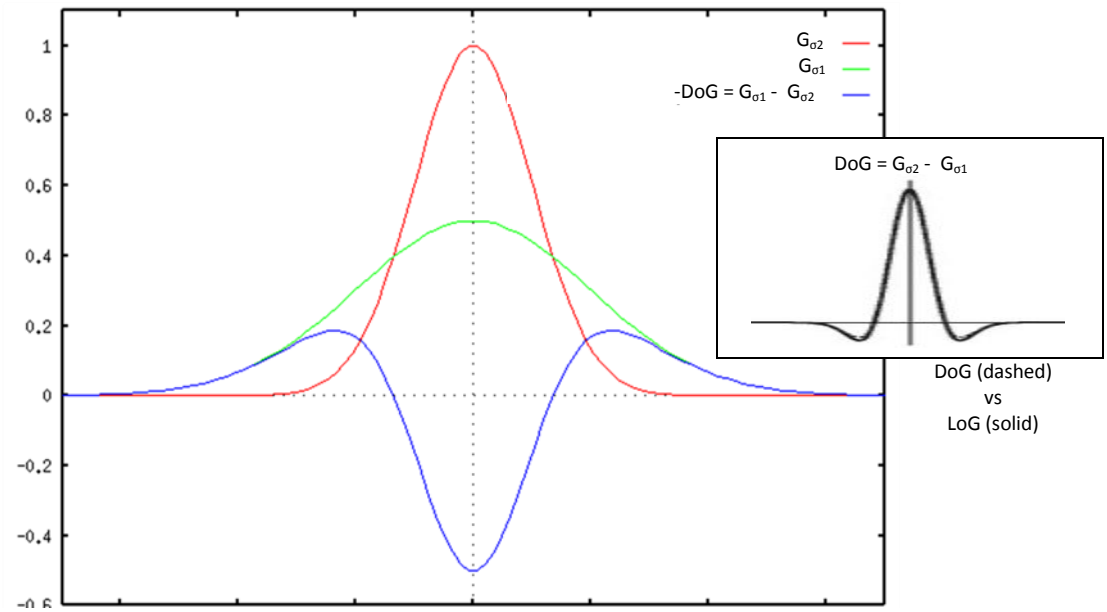


Figure 2-7 The 1D visualization of the inverted DoG as the result of subtracting two Gaussian kernels of different widths (Drakos and Moore 2007). The inset graphic shows little if any perceivable difference between the LoG and DoG convolution kernels (Gonzalez and Woods 2007).

The DoG can be seen as the 1D difference between the two Gaussian kernel widths (Drakos and Moore 2007) and is then compared to the Log Filter in the inset of Figure 2-7 (Gonzalez and

Woods 2007). So at this point, it is possible to extract distinct features from the edge detail within an image. In fact, by utilizing the LoG and DoG kernels it is possible to accentuate and identify the best edge detail and from these regions extract robust invariant features from a scene.

2.2 Matching Invariant Features

The following technique, which is utilized for matching corresponding features, was originally utilized in astronomy to register images of "star fields" (Chandrasekhar 1999). Since the LoG filter can be utilized to reduce an image to repeatable point sources, the author was able to successfully implement the same approach to properly filtered terrestrial images.

The accuracy of registering images utilizing the LoG technique boils down to how well related areas of both images can be identified, isolated, and matched. Even though the LoG threshold procedure simplifies the registration process by reducing the images to point sets. It is the accurate matching of points, from dissimilar point sets, that will determine the utility and ultimate success of most registration processes.

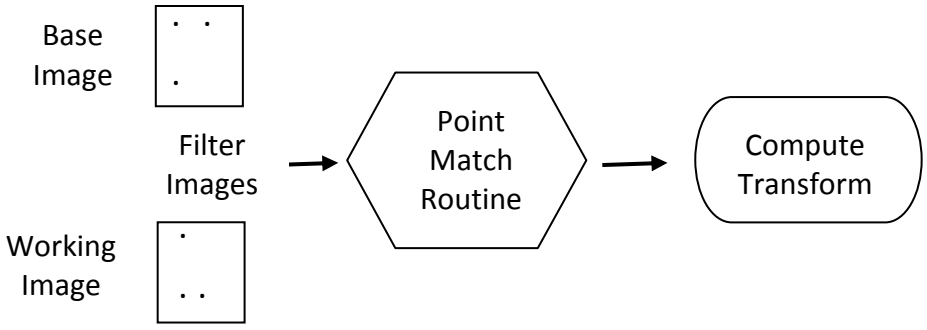


Figure 2-8 Matching points to determine the Image Transform.

Throughout the next two sections, robust point matching techniques are introduced and applied to the task of image registration. An important concept to keep in mind is that the matched points will provide the matrix equation inputs to solve for the geometric relationship between two images. So, if an image is shifted, rotated, and scaled with respect to (w.r.t.) a reference image, then we require three sets of matched points (6 equations) to solve for the 5 DOF required to register this image pair. If we have more matched points than required, the solution is over-determined and it is possible to either select a subset of the "best" point matches that uniquely determine the solution or utilize a linear regression model to estimate the best fit to the data and obtain subpixel registration accuracy.

2.2.1 Point Matching using Distance Similarity

This process utilizes a point's distance from every other point in a scene and creates an array of distances with this data. This is done with each point in the image, from which a matrix of distances is created. The point distance matrices, from each image, are then compared row-to-row for the total number of matching distances. The two rows that have the greatest number of distance matches (within some designated error) are considered matched points as shown in Figure 2-9.

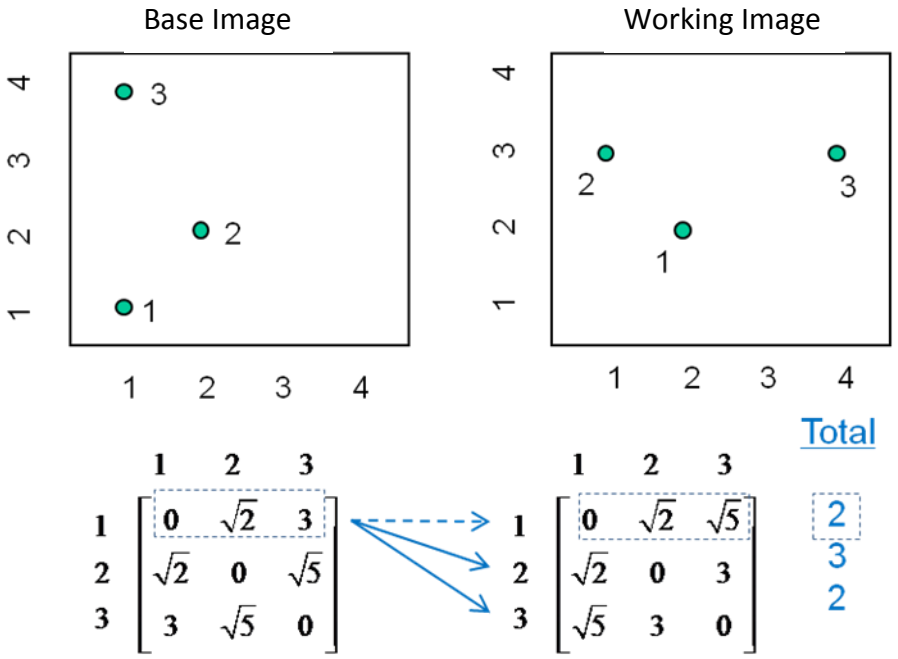


Figure 2-9 Determining matching points through equivalent distances to other points.

The distance between any two points is equal to the square root of the sum of the squares: Distance, $d = \sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$. For our reference image points 2 and 3, this becomes: $d = \sqrt{(2-1)^2 + (2-4)^2} = \sqrt{5}$. In the matrix, each row and column represents that point's distance from the other points, which are also related to their equivalent row and column. In our example above, point-1 from the reference image would match point-2 from the warp image, since they have the greatest number of matching distances in their equivalent rows and columns.

Additional similarity metrics can also be imposed to compare the relative relationship of a feature to its proposed match, in order to cull bad matches. Angle relationships were introduced by utilizing a 3D matrix comparison of vertex angles. Additionally, the normalized

LoG maxima and minima are compared to help discriminate features and mitigate the effects of illumination variation.

Scale invariance can be established by comparing the ratio of point distances to every other point or through the use of multi-scale techniques, such as image pyramid (Wavelet) analysis (Walli, Multisensor Image Registration utilizing the LoG Filter and FWT 2003). Additionally, scale effects can be addressed directly in the filter itself by implementing a scale normalized version, where $LoG = \sigma^2 \nabla^2 f$ (Lindeberg 1994).

Finally, projective invariance can be addressed through the comparison of the cross ratio of distance ratios. This cross ratio, of four collinear points, is the most fundamental projective invariant and can be visualized below (Hartley and Zisserman 2004) and (Kraus, Harley and Kyle 2007).

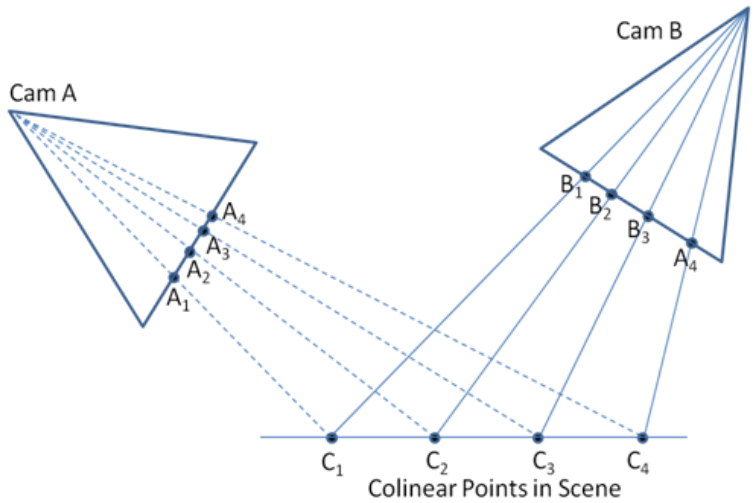


Figure 2-10 Each set of points has the same cross ratio and are related via line-to-line projectivity (Hartley and Zisserman 2004).

2.2.2 Point Matching using Localized Gradient Similarity

The Shift Invariant Feature Transform (SIFT) operator (D. G. Lowe 2004), has become a "gold standard" in 2D image registration due to its ability to robustly identify large quantities of semi-invariant feature within images. Whereas the author's LoG and Wavelet Registration (LoGWaR) technique could produce hundreds of extracted GCPs per image, the SIFT technique can produce thousands on images of comparable size. This is extremely useful when attempting to create sparse structure from matched point correspondences. In addition, more recent independent testing has confirmed that the SIFT feature detector, and its variants, perform better under varying image conditions than other current feature extraction techniques (Moreels and Perona 2006) (Mikolajczyk and Schmid 2005).

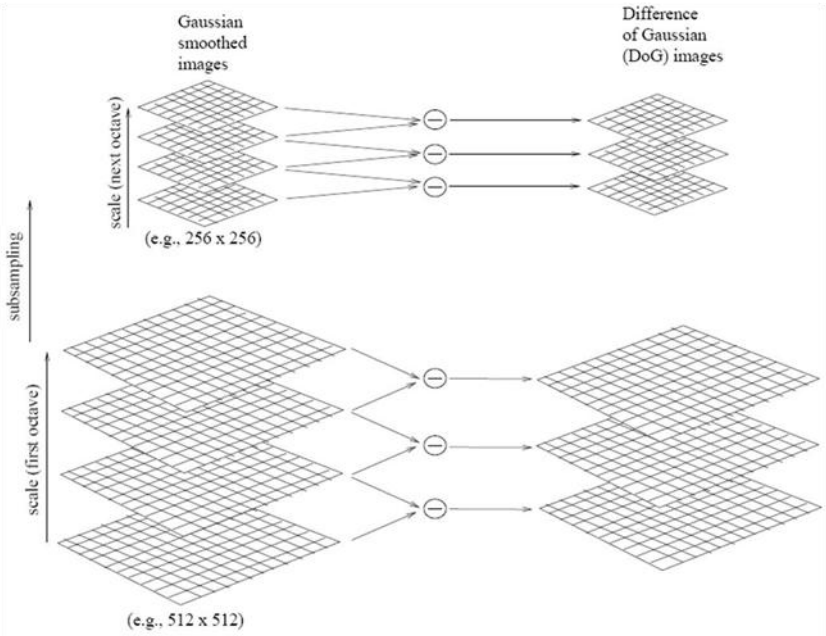


Figure 2-11 For every octave of scale space the initial image is convolved with Gaussians of varying standard deviations and subtracted from their neighbors producing a DoG pyramid (D. G. Lowe 2004).

Figure 2-11 - Figure 2-13, portray the basic approach that the SIFT algorithm uses for feature

extraction (D. G. Lowe 2004). The process begins by filtering the image with Gaussians of varying standard deviation across a given image scale, where $\sigma_{i+1} = \sqrt{2}\sigma_i$ By varying sigma by a constant value across an octave, Lowe was able to show mathematically the equivalence of this filter to the scale normalized LoG. These smoothed images are then subtracted from each other to extract the edge detail at varying spatial frequencies, thus giving the technique its name "Difference of Gaussian". This is then repeated at each image octave (dyadic power), where the image is decimated (scaled in half) to some arbitrary fraction of the former image dimension.

The next step is to extract the maxima and minima keypoints from the filtered images. This is accomplished by comparing each sample point to its neighbors in the same filter image and its scale neighbors that will have extracted slightly difference spatial frequencies due to the Gaussian width changes induced by varying sigma.

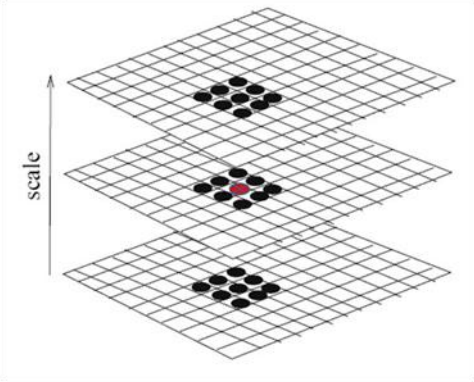


Figure 2-12 Maxima and Minima of the DoG pyramid stacks are detected by comparing each pixel with its 26 neighbors in 3x3 regions at the current and adjacent scales (D. G. Lowe 2004).

Each location is selected as a minimum or maxima only if it is the largest or smallest among

these neighbors, as shown in Figure 2-12. Lowe argues that the cost of checking every location is acceptable since most sample points will be eliminated after the first few checks.

Figure 2-13, shows how SIFT maps out the gradients of the surface surrounding the keypoint locations. In this example, each 4x4 subregion is described as an 8 element orientation histogram, where the individual gradient magnitude is added to the "closest" bin.

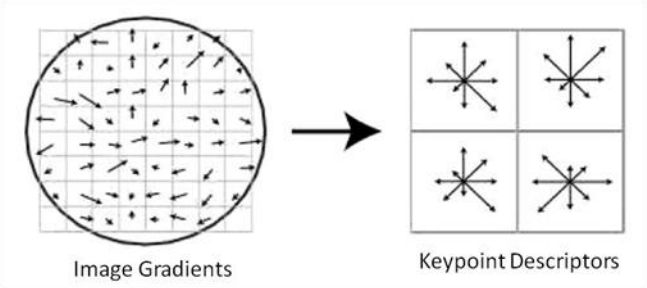


Figure 2-13 Keypoint descriptors are created by computing the gradient magnitude and orientation, Gaussian weighted by the pixels location, surrounding a keypoint. These samples are then accumulated into 8 bin orientation histograms, which summarize a 4x4 subregion (D. G. Lowe 2004).

While the example above (Figure 2-13) only shows an 8x8 element analysis around a keypoint, the actual algorithm observes a 16x16 region. This regional mapping is then stored into a 128 element vector (4x4x8) of orientation histograms that can be utilized to compare against the regional descriptions of keypoints in other images. Lowe refers to the closest histogram vectors in different images as "nearest neighbors" and assigns them as a potential match. If these descriptors are then normalized, they can be quite robust against the effects of scene illumination (D. G. Lowe 2004).

A demonstration of the robust, invariant feature detection possible with the SIFT algorithm is available in Figure 2-14. In this example, thousands of keypoints were generated on two 1kx1k images of the VanLare site to create hundreds of good matches for developing a precise

registration transform. It is easy to see the general flow of the correspondences from one image to the next and to visually detect outliers that deviate from the norm.

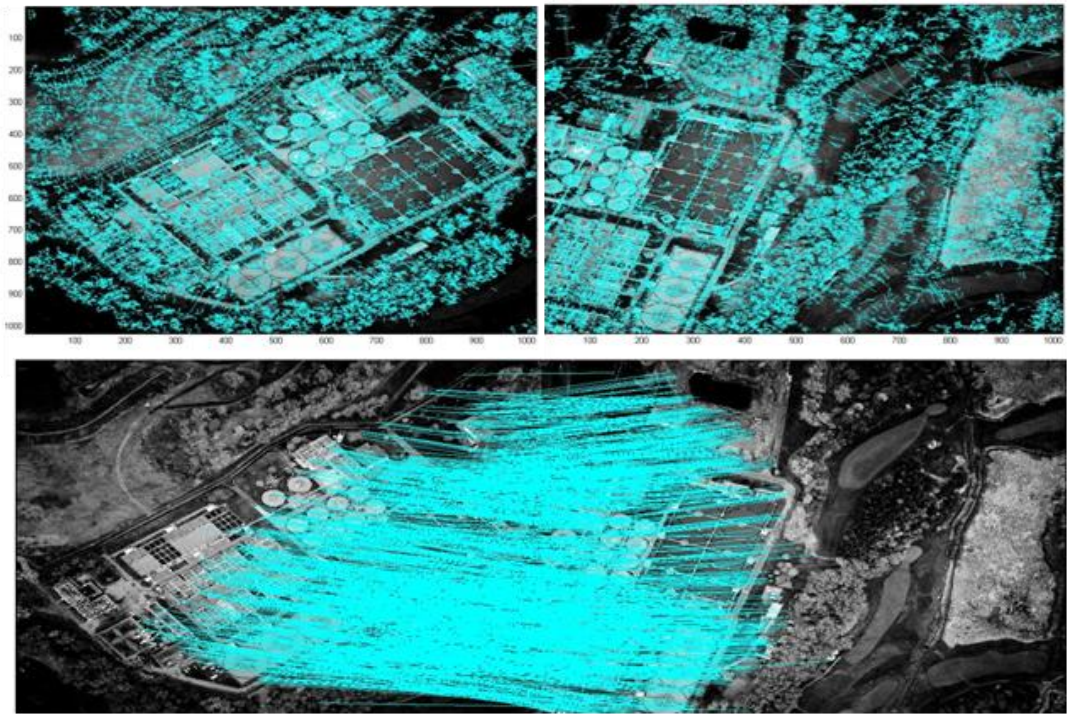


Figure 2-14 Thousands of invariant keypoints generated and matched using the SIFT algorithm.

Lowe maintains that the best candidate match for each keypoint will be the one that has the minimum Euclidean distance from the invariant descriptor vector under analysis. A simple way to compare the minimum Euclidean distance of description vectors is to take the dot product of two vectors to gauge their similarity as a potential match. This technique is very similar to the common spectral signature comparison algorithm called Spectral Angle Mapper (SAM).

Since some descriptors will not have any "good" match, because they were not detected in the other image, it is necessary to devise a technique to cull outliers early in the process. An effective method is to compare the distance of the closest match, to that of the second closest. This measure performs well because an actual correspondence will often have their closest

potential match much closer, relative to the second closest, than an incorrect match. False matches will often have several other matches that are relatively close due to the high dimensionality of the feature space. Lowe found that rejecting all matches with a closest-to-next closest ratio of 0.8 would eliminate ~90% of the bad matches while eliminating only ~5% of the good matches (D. G. Lowe 2004).

2.3 Transform Development

Once a valid set of correspondences, or matched GCPs have been obtained via automated or user assisted means, it is possible to utilize these points to develop a transform to warp the working image into the spatial domain of the base image. This polynomial expression is covered in several pieces of literature (Schott 2007) and (Schowengerdt 2007), and takes the following general form of (9) & (10), where [x, y] represents the warp image coordinates and [X, Y] represents the base image coordinates.

$$X_i = \sum_{k=0}^{N} \sum_{l=0}^{N-1} a_{kl} x^l y^k + \varepsilon_x \quad \textbf{(9)}$$

$$Y_i = \sum_{k=0}^{N} \sum_{l=0}^{N-1} b_{kl} x^l y^k + \varepsilon_y \quad \textbf{(10)}$$

This section will utilize a subset of the general polynomial expressions, both 1st order and affine. The affine coefficients are the linear relationships that allow for shift, scale, rotation and skew between two images of interest and are represented by the first 3 terms in the polynomial expressions below in (11) and (12). The last, multiplicative term, completes the 1st

order polynomial expression with a coefficient which enables a projective transformation from the warp image to the base image domain.

$$X_i = a_{00} + a_{10}x_i + a_{01}y_i + a_{11}x_i y_i \quad \textbf{(11)}$$

$$Y_i = b_{00} + b_{10}y_i + b_{01}x_i + b_{11}x_i y_i \quad \textbf{(12)}$$

This 1st order polynomial expression can be put into a compact 3x3 matrix notation which is convenient for mathematical manipulation as is evident in (13),

$$\begin{bmatrix} a_{01} & a_{10} & a_{00} \\ b_{01} & b_{10} & b_{11} \\ a_{11} & b_{11} & 1 \end{bmatrix} = \begin{bmatrix} \mathrm{h}_{11} & \mathrm{h}_{12} & \mathrm{h}_{13} \\ \mathrm{h}_{21} & \mathrm{h}_{22} & \mathrm{h}_{23} \\ \mathrm{h}_{31} & \mathrm{h}_{32} & \mathrm{h}_{33} \end{bmatrix} = H \quad \textbf{(13)}$$

where H is the homogeneous 2D image transform (homography), that relates the warp image to the base image. In order to solve for a projective transformation, the warp image coordinate would take the following forms (Hartley and Zisserman 2004), shown in (14) and (15).

$$X_i = \frac{h_{11}x_i + h_{12}y_i + h_{13}}{h_{31}x_i + h_{32}y_i + h_{33}} \quad \textbf{(14)}$$

$$Y_i = \frac{h_{21}x_i + h_{22}y_i + h_{23}}{h_{31}x_i + h_{32}y_i + h_{33}} \quad \textbf{(15)}$$

The ability to relate images utilizing a matrix transformation approach is extremely useful and is covered very well in “Digital Image Warping” (Wolberg 1990). By utilizing a homogeneous coordinate system to represent the points and transformation allows us to linearize the solution for least squares analysis. To implement a homogeneous coordinate system, we essentially add another dimension to the image point and transform descriptions. This can be

accomplished by the following mathematical representations for the reference (base) image locations $\vec{X}$ and working (warp) image locations $\vec{x}$ (Wolberg 1990).

Working

$$\vec{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \textbf{(16)}$$

Base

$$\vec{X} = \begin{bmatrix} X \\ Y \\ 1 \end{bmatrix} \quad \textbf{(17)}$$

Rotation

$$R_H = \begin{bmatrix} cos\theta & sin\theta & 0 \\ -sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(18)}$$

Scale Transform

$$S_H = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(19)}$$

Translation Transform

$$T_H = \begin{bmatrix} 1 & 0 & \mathrm{t_x} \\ 0 & 1 & \mathrm{t_y} \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(20)}$$

Composite Transform

$$H = RST = \begin{bmatrix} s_x cos\theta & s_y sin\theta & \left(t_x s_x cos\theta + t_y s_y sin\theta\right) \\ -s_x sin\theta & s_y cos\theta & \left(t_y s_y cos\theta - t_x s_x sin\theta\right) \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(21)}$$

Simplified Notation

$$\vec{X} = H\vec{x} \quad \textbf{(22)}$$

Matrix Formula

$$\begin{bmatrix} X_1 & X_2 & \cdots X_i \\ Y_1 & Y_2 & \cdots Y_i \\ 1 & 1 & \cdots 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & 1 \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots x_i \\ y_1 & y_2 & \cdots y_i \\ 1 & 1 & \cdots 1 \end{bmatrix} \quad \textbf{(23)}$$

Inverse Solution

$$H = \vec{X}\vec{x}^{-1} \tag{24}$$

Pseudo-Inv Solution

$$H = \vec{X}\vec{x}^{T}\left(\vec{x}\vec{x}^{T}\right)^{-1} \tag{25}$$

Simplified Notation

$$H = \vec{X}\vec{x}^{\dagger} \tag{26}$$

Where, $\vec{X}\vec{x}^{\dagger}$ represents the Psuedo Inverse Least Squares solution to H. The five rotation, scale, and translation parameters can be extracted by utilizing the following equations,

Rotation Angle

$$\theta = tan^{-1}\left(\frac{H_{12}}{H_{22}}\right) = atan2(H_{12}, H_{22}) \tag{27}$$

Scale x-axis

$$S_{x=}\frac{H_{11}}{cos\theta} = \sqrt{H_{11}^2 + H_{21}^2} \tag{28}$$

Scale y-axis

$$S_{y=}\frac{H_{12}}{sin\theta} = \sqrt{H_{12}^2 + H_{22}^2} \tag{29}$$

Translation x-axis

$$t_{x=}\frac{H_{13}cos\theta - H_{23}sin\theta}{S_x} \tag{30}$$

Translation y-axis

$$t_{y=}\frac{H_{13}sin\theta + H_{23}cos\theta}{S_y} \tag{31}$$

Although care must be taken to avoid division by zero with some of these solutions, alternate equations can be obtained when necessary.

2.4 Constraining the Transform Results – 2D Conformal and Affine

Due to the construction of the homogeneous coordinates and 2D homography in matrix notation, it is often easier to solve for the full projective solution than it is for an affine or conformal (rigid body) transformation. This is in part due to the ease of solving for the 3x3 projective through the linear least squares (pseudo-inverse, Eq 2-22) method. However, many times, we will want to constrain the solution to the transformation that changes the data least and still relates them properly. This minimalist approach is not only relevant for 2D image registration, but, 3D structure registration as well (Section 5.4). Since it is often beneficial to induce only the rigid body effects of rotation, scaling, and translation (RST), the following Direct Linear Transform (DLT) approach can be utilized to solve for the 4 unknown parameters (R, S, T_x, T_y) of the 2D conformal transformation (DeWitt and Wolf 2000).

$$\vec{X} = RS\vec{x} + T \tag{32}$$

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} S & 0 \\ 0 & S \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix} \tag{33}$$

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \begin{bmatrix} h_1 & -h_2 \\ h_2 & h_1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} h_3 \\ h_4 \end{bmatrix} \tag{34}$$

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \begin{bmatrix} x_i & -y_i & 1 & 0 \\ y_i & x_i & 0 & 1 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \end{bmatrix} \tag{35}$$

The least squares solution can be obtained through the pseudo-inverse technique (26). It is useful to note that the h_1 and h_2 coefficients are not only part of the rotation matrix, but, also contain the rigid body scaling component of the conformal transform $(S = S_x = S_y)$.

In a similar manner, The DLT technique can also be utilized to solve for the six unknowns of the affine transform (R, Sx, Sy, Tx, Ty, and either k_x or k_y), where shear parallel to the x-axis results in $x' = x + k_x \times y$ and shear parallel to the y axis delivers $y' = y + k_y \times x$.

$$\vec{X} = RSW\vec{x} + T \tag{36}$$

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \begin{bmatrix} cos\theta & -sin\theta \\ sin\theta & cos\theta \end{bmatrix} \begin{bmatrix} S_x & 0 \\ 0 & S_y \end{bmatrix} \begin{bmatrix} 1 & k_x \\ k_y & 1 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \end{bmatrix} \tag{37}$$

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \begin{bmatrix} h_1 & h_3 \\ h_2 & h_4 \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} h_5 \\ h_6 \end{bmatrix} \tag{38}$$

$$\begin{bmatrix} X_i \\ Y_i \end{bmatrix} = \begin{bmatrix} x_i & 0 & y_i & 0 & 1 & 0 \\ 0 & x_i & 0 & y_i & 0 & 1 \end{bmatrix} \begin{bmatrix} h_1 \\ h_2 \\ h_3 \\ h_4 \\ h_5 \\ h_6 \end{bmatrix} \tag{39}$$

2.5 Outlier Removal and Error Analysis

Once the initial matched point set has been obtained by automated means, it will always be necessary to test for bad matches or "outliers". The following two methods offer robust outlier removal, but, are fundamentally different in their conception. The statistical RMS Distance Error (RMSDE) technique utilizes the weight of all of the matches to estimate a solution and removes those matches with the most error or those that vary by some standard deviation from the mean. Alternatively, the RANdom SAmple Consensus (RANSAC) algorithm (Fischler

and Bolles 1981) can be utilized to robustly remove outliers from the data and will be discussed in greater detail in Section 2.5.2.

2.5.1 RMSDE Analysis

The RMSDE metric computes the deviation from a polynomial model to determine registration accuracy. RMSDE, is one of the more common techniques utilized in remote sensing for judging the "goodness" of a registered dataset. In fact, the RMSDE technique is even used by ENVI to judge deviation of matches from the prescribed polynomial model to judge registration accuracy. Discriminating outliers based on deviation from a mathematical model describing the transform from one image domain to another is shown in Figure 2-15. By analyzing the error associated with each matched point from the polynomial model of choice, it is possible to reject bad matches.

One way to do this is through analysis of the standard deviation from the RMSDE. Any matches that deviate significantly from the mean (> 1 STD) can then be removed. If additional iterations are required to derive a transform within a given error constraint, the matches below a given threshold could be removed based on their deviation from the mathematical model.

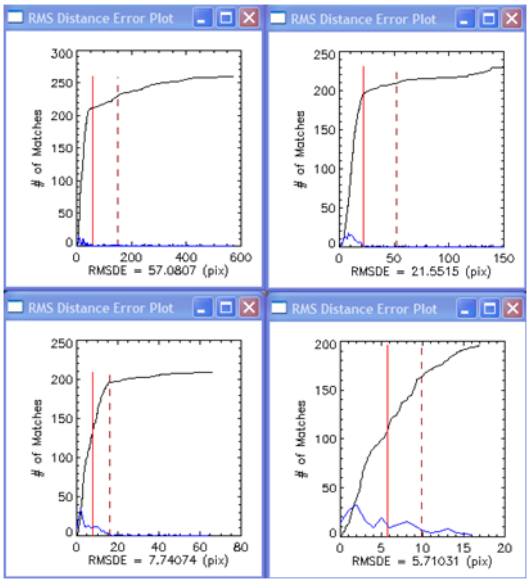


Figure 2-15 Utilizing RMSDE as a Metric to cull Outliers; note the distinctive "knee" in the error curve.

This can be done iteratively to determine a statistical solution that is of low enough error to satisfy the accuracy of registration required for a given task. Figure 2-15 shows this process, which utilizes an iterative statistical solution to cull the outliers. Note the distinctive "knee" in the curve of the error plot; is a good indicator of the presence of outliers. The iterative pruning of match points can deliver a registration with subpixel accuracy. In fact, it is an easy task to continually remove the match with the greatest error, until the total RMSDE is less than a user defined quantity. Obviously, one would like to maintain a significant number of points relative to the degrees of freedom while still ensuring that the matched locations encompass as much of the two images as possible.

2.5.2 Random Sample Consensus (RANSAC) Analysis

The RANSAC technique iteratively and randomly samples the minimal amount of matches required to develop a given mathematical relationship. Once this is done, it determines the

number of inliers and outliers from that relationship using prescribed error thresholds. After a statistically meaningful number of samples have been taken, it will remove outliers based on the best (most inliers) model that was derived. Figure 2-16, graphically portrays this robust technique (Hartley and Zisserman 2004).

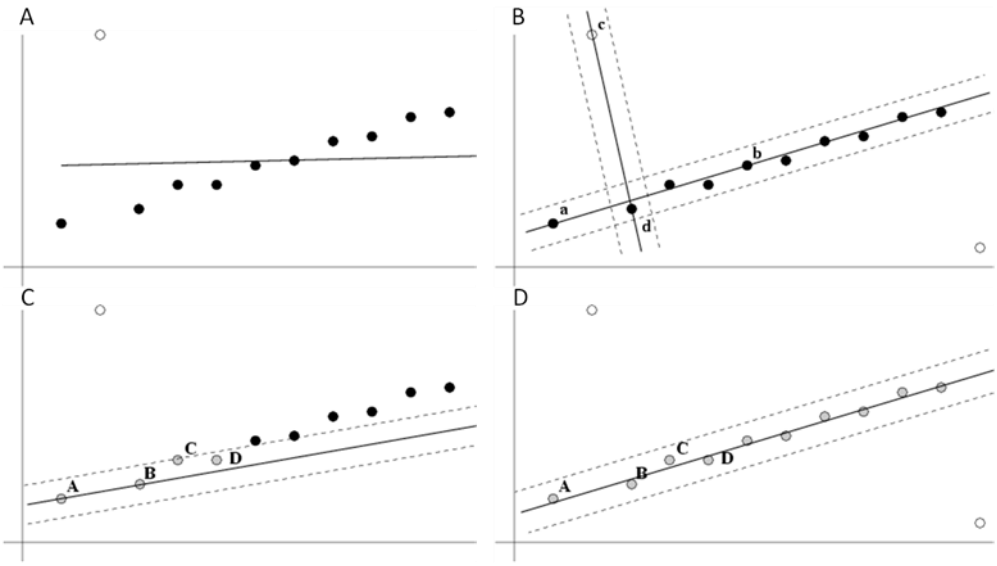


Figure 2-16 A) A dataset with outliers; B) Shows how a line can be determined with the minimal number of two points and how the inliers are tallied; C) Shows how two close points can provide poor extrapolation and low inlier count; D) Shows the "correct" solution for culling the outliers.

RANSAC has proven to be a robust technique for outlier removal, even in the presence of large numbers of incorrect matches. Also, because it is not necessary to test all the sets of points for a solution, it can be efficiently utilized with techniques like SIFT that provide large numbers of automated matches. This technique will be covered in greater detail starting in Section 4.1.

For most of the 2D image-to-image registration and outlier removal, the SIFT algorithm will be utilized in conjunction with RANSAC, unless otherwise noted, due to their robust performance under various imaging conditions. In the next section we will increase the dimensionality of one of the datasets in order to relate images with 3D models.

3 Relating Images to Models

Since 2D registration will always be limited by the effects of projective viewing geometry, occlusion, shadowing, terrain elevation and building height variations, it is essential to model the 3D influences on a scene so that they can be adequately mitigated. If a 3D model of the target scene is available, it is possible to orient the model to the same viewing perspective as the camera that acquired the image and then project it onto the same 2D plane as the target image. Once this is accomplished with enough accuracy, traditional 2D registration techniques can be utilized to relate the image to the projected model. Essentially, the 3D ambiguity between the model and the image are removed and the image can then be utilized as a texture map on the model. If this is done properly, the 3D nature of the image that was lost when the image was acquired can be substantially regained.

3.1 Known Camera Pose

This approach relies on knowledge of the camera pose (position and orientation), to estimate the proper 3D scene projection relative to the remotely sensed image. Once the initial model orientation is estimated, this knowledge can be utilized with scene based registration to properly overlay imagery within Geographical Information System (GIS) applications, such as Google Earth (Walli and Rhody, Automated Image Registration to 3-D Scene Models 2008).

This technique can allow a user to properly place imagery within a 3D environment using simple geographic location descriptions that can be coded in script languages, like the Keyhole Markup Language (KML). Additionally, projected imagery from the camera acquisition location has

been implemented with the AANEE program (Section 1.3) as a technique to blend various modalities of interest (i.e. Pseudo-Color IR) as shown in Figure 3-1 below.

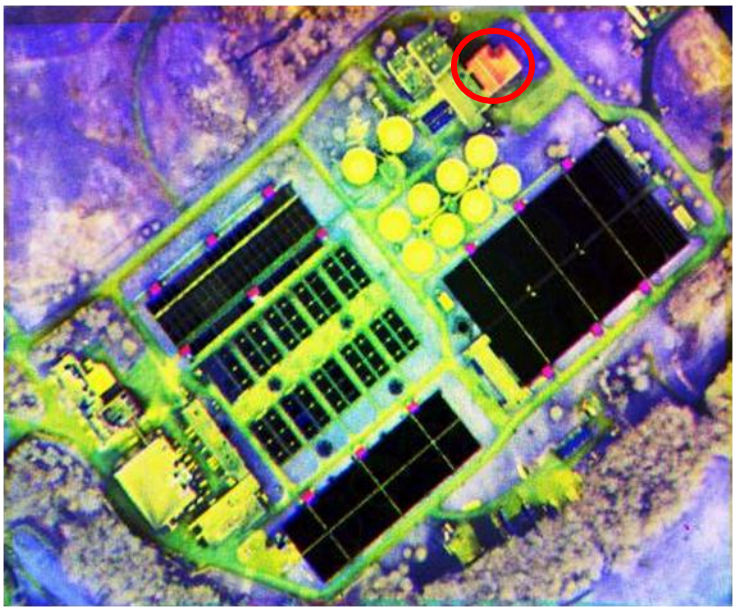


Figure 3-1 In this Pseudo-Color composite of the WASP SWIR/MWIR/LWIR composed as an RGB image stack, the Northern Bldg at the VanLare Plant (Red Circle) was recently built and is evidently made of a different material than its neighbors.

The process in Figure 3-2, describes the basic steps required to solve for precise image-to-model registration when the camera pose is known. This step can be implemented as a way to mitigate any residual error in the accuracy of the sensor's Inertial Measurement Unit (IMU) pointing and Global Positioning System (GPS) location parameters.

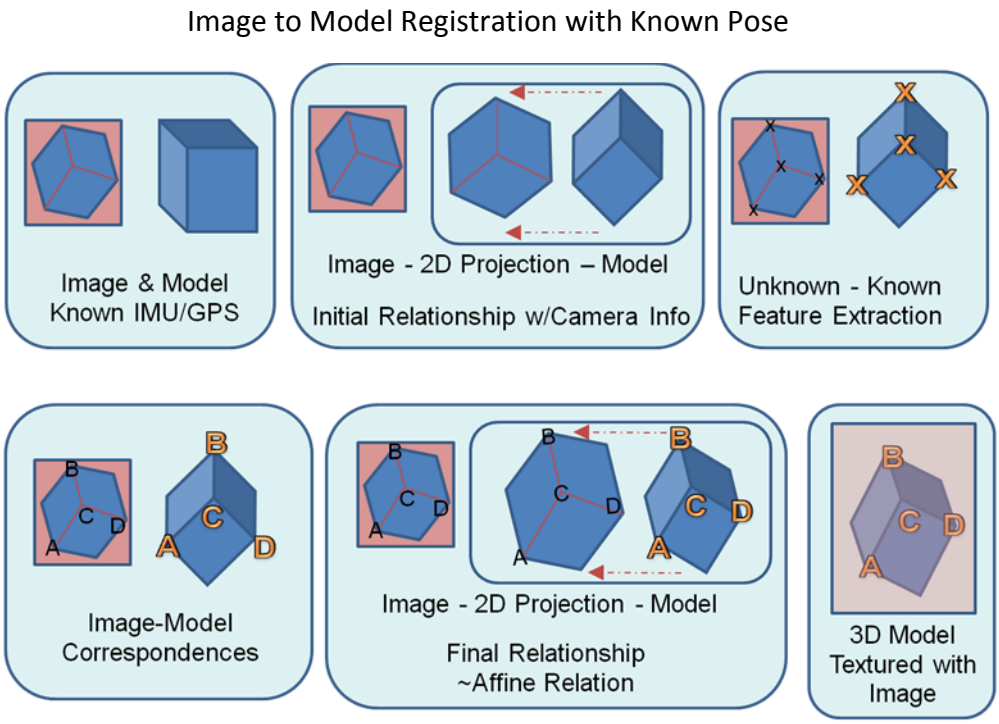


Figure 3-2 The process for relating an image to a model when the camera pose is known starts with changing the orientation of the model to mimic the known sensor view. Then the extraction and matching of similar features from the image and model can occur in similar 2D construct. These matches are then used to refine the model pose (due to IMU/GPS precision error) for final projective texturing of the image on the model.

3.1.1 Approach

For this section we assume that the camera pose is available and that we have a model that has been textured with imagery of a similar modality. Since this approach primarily focuses on removing the 3-D ambiguity of the registration process, it should be applicable to most automated 2-D image registration techniques if even rudimentary models of a scene are available. Additionally, since all remotely sensed images are influenced by the 3-D world in some manner, it is important to understand and control these effects whenever possible. The utility of relating these two datasets should be readily apparent from Figure 3-3. Here a crude model has been textured (by draping an image-Section 15.1) and oriented to the acquisition view of another image to remove some of the undesirable 3D influences.

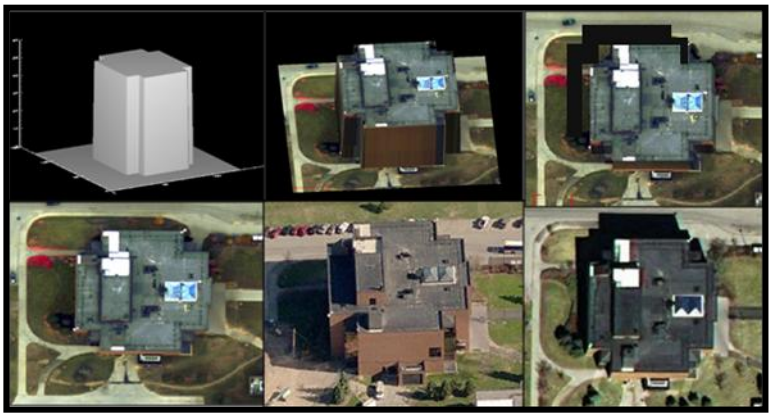


Figure 3-3 Even rudimentary models textured with images (top) can be used to simulate the 3D effects of scene projection, shadowing, and occlusion evident within real images (bottom) and can thus allow for precise 2D registration.

Since we can orient our model to the respective orientation dictated by the known camera pose, it is possible to project our model into a similar space to that of the target image and utilize traditional point extraction techniques for feature extraction. Additionally, since we have assumed that the model is textured with imagery of a similar modality, the extracted edge detail should provide similar features from both the projected model and target image.

With powerful physics based modeling software, like Digital Imaging and Remote Sensing Image Generation (DIRSIG), it is possible to replicate the appearance of many modalities including visible, infrared, polarimetric, synthetic aperture radar, and low light panchromatic (Digital Imaging and Remote Sensing Laboratory 2006). This increases the probability of extracting similar features from a wide range of potential modalities, since the edge detail should be accurately generated if the model is geometrically correct and removes the 3D ambiguity.

The feature matching in this section will be approached in the same way as traditional 2D feature matching in Section 1.2. However, when utilizing a GIS environment such as Google

Earth (Google Earth 2010), care must be taken to ensure that the modeled scene contains the appropriate terrain and building models, but that the working image is unaltered. In this way, when the pose of the acquisition camera has been properly encoded into the viewport, the working image should closely resemble the modeled scene (Walli and Rhody, Automated Image Registration to 3-D Scene Models 2008).

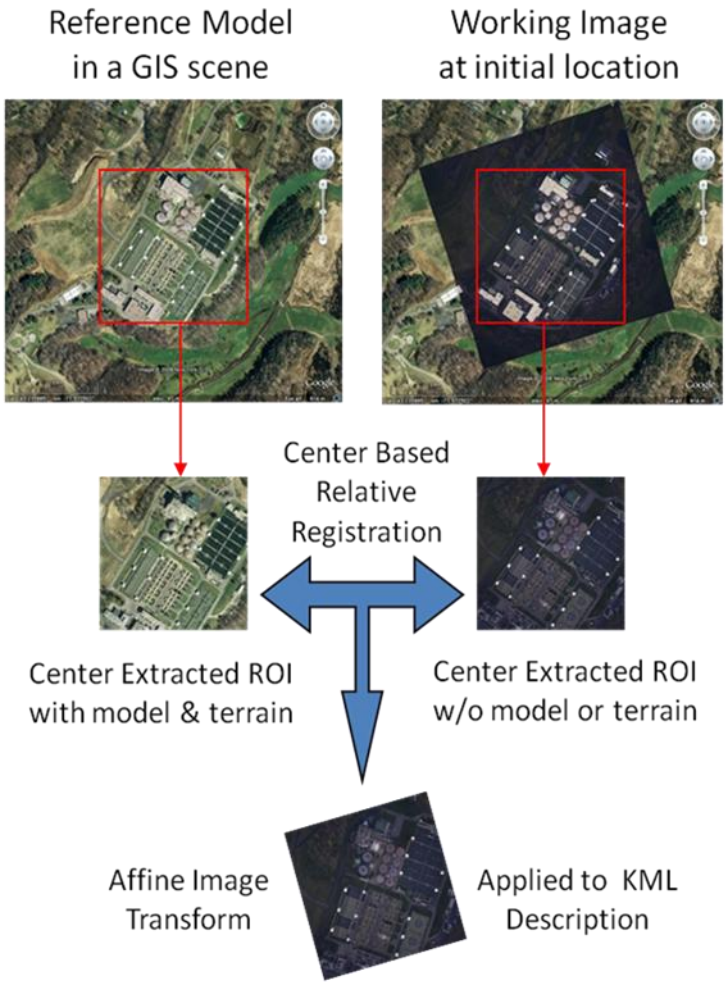


Figure 3-4 The general process utilized to register images to GIS modeled scenes.

If the IMU/GPS information is sufficiently accurate, then a 2D affine relationship will often provide acceptable accuracy to relate the model projection to the target image. This approach provides a piece-wise affine solution to the projective problem presented by our camera pose estimation. It can be likened to a linear estimate of a nonlinear problem that provides acceptable levels of error over small regions of the solution space and is shown in Figure 3-4, where a hi-fidelity model of the VanLare site was provided by Pictometry International Corporation (Pictometry 2010).

3.1.2 Case Study - Using Google Earth Models and WASP Imagery

This section provides a brief summary of results obtained when implementing the procedure outlined above. For this case study, Google Earth (GE) was utilized as the GIS visualization tool, with detailed Collada models of the Frank E. VanLare Water Treatment Plant embedded within the standard satellite imagery and 30[m] terrain elevation maps. The working imagery was obtained from RIT's Wildfire Airborne Sensing Program's (WASP) multimodal sensor suite that provides 4kx4k Visible Near Infrared (VNIR) and 640x512 Short Wave Infrared (SWIR), Mid-Wave Infrared (MWIR), and Long Wave Infrared (LWIR) images.

A significant limitation in utilizing GE's as the GIS for 3D scene representation, is that terrain overlaid image descriptions are limited to heading angle and a Latitude/Longitude box. This can limit the transformations to those of an affine nature if preprocessing of the imagery is not performed. Additionally, this tool was designed for square North/South and East/West "box" areas, and so the working imagery was designed for ortho-rectification as a requirement for

proper implementation. After working through some of GE's limitations, the results, seen in Figure 3-5 and Figure 3-6, show that precise registration is available by utilizing this technique.

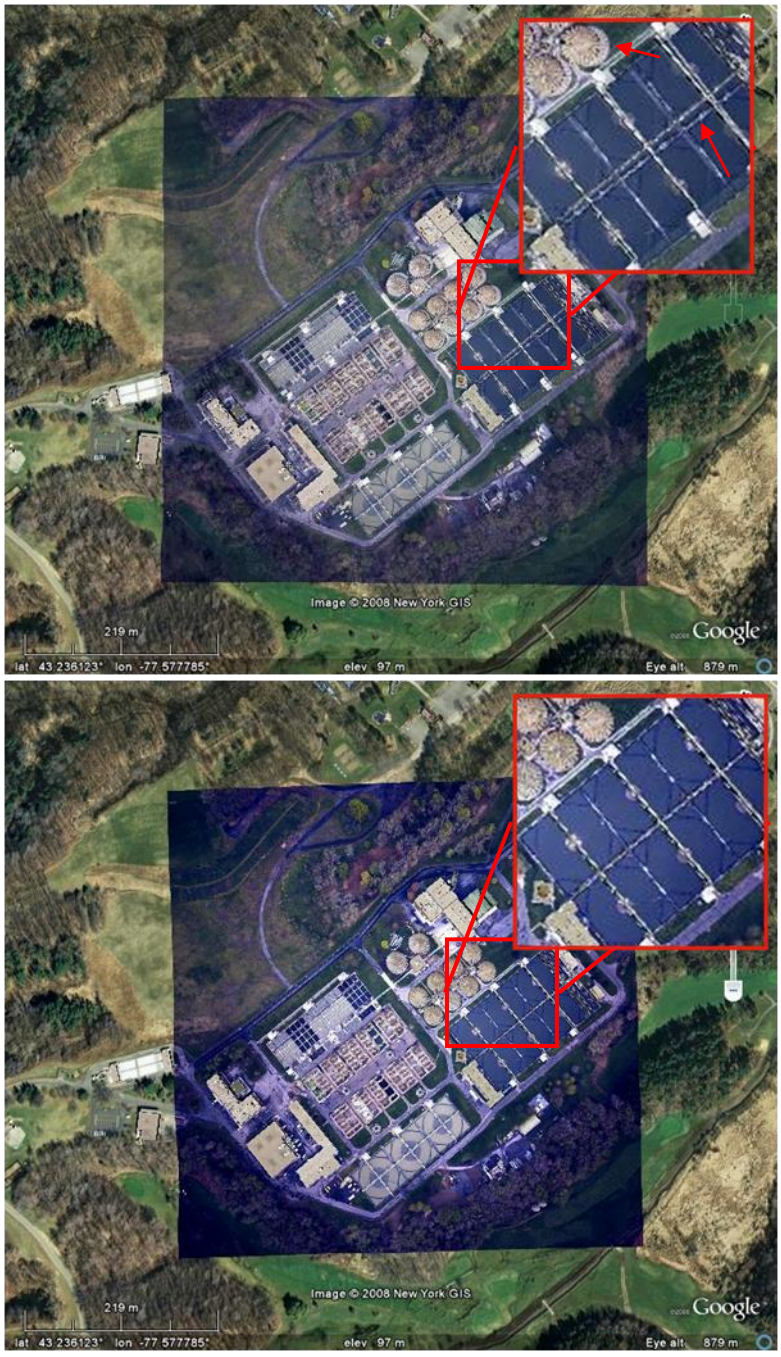


Figure 3-5 The top image with initial IMU/GPS pose and the bottom after affine correction. Both images are displayed in Google Earth with 30m accuracy terrain and detailed Pictometry model of the VanLare Site.

The careful observer will note the displacement of building models from their placement w.r.t. the imagery in Figure 3-5, especially over the settling ponds in the SE area of the VanLare Plant. In the multimodal results of Figure 3-6 the registered image is overlaid on top of the initial IMU/GPS location. This shows the relative placement and correction obtained from in-scene registration compared to the initial hardware solution.

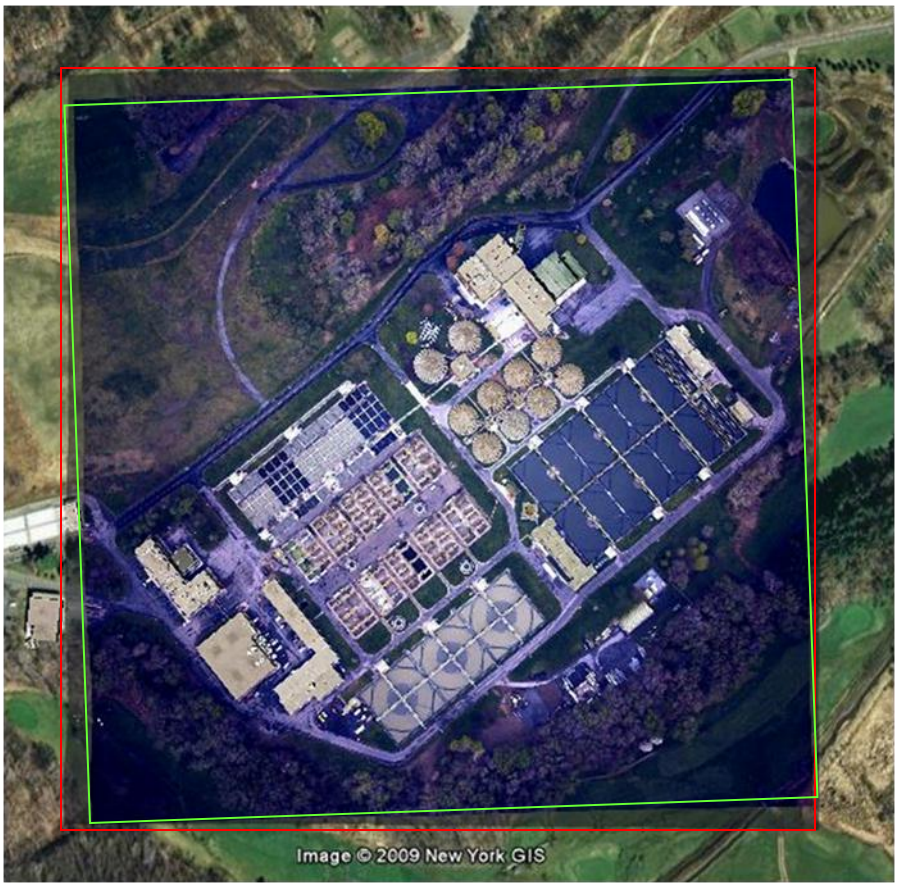


Figure 3-6 Comparison of Registered VNIR WASP image (outlined in green) overlaid on its initial location (outlined in red) with the detailed site model in GE.

In order to overcome the GE limitations for texturing the terrain with imagery, it is essential to implement mathematical techniques that link the camera orientation parameters to the 2D Projective Homography (Seedahmed 2006). This technique requires that a true planar

relationship exists between the correspondences used to create the projective transformation. While this may not always be the case, the statistical techniques developed by the author in Sections 2.5.1 and 6.4.1.1 to ensure accurate RMSDE consistency of the model with the match points can be utilized to constrain the solution space. This is particularly relevant and applicable to the Section 6.4.1.1, where a 2D planar relationship between the image and model is warranted due to accurate scene modeling.

3.2 Unknown Camera Pose

Given the utility of using camera information to remove 3D ambiguity from a registration result, the next logical task is to determine this camera information from "in-scene" information when it is not available. The ability to estimate the position and orientation of a camera (camera pose), without prior knowledge, is often essential for relating imagery of a given scene. The ability to use known 3D control points and corresponding image locations to retrieve camera position and orientation when the image was acquired is referred to by photogrammetrists as "space resectioning". Resectioning images can be a very powerful technique, since the camera information for a given image of interest, may not be readily available. Even when it is, the accuracy of that information may be unknown or may not be precise enough to use without further refinement. This resectioning process is shown in Figure 3-7.

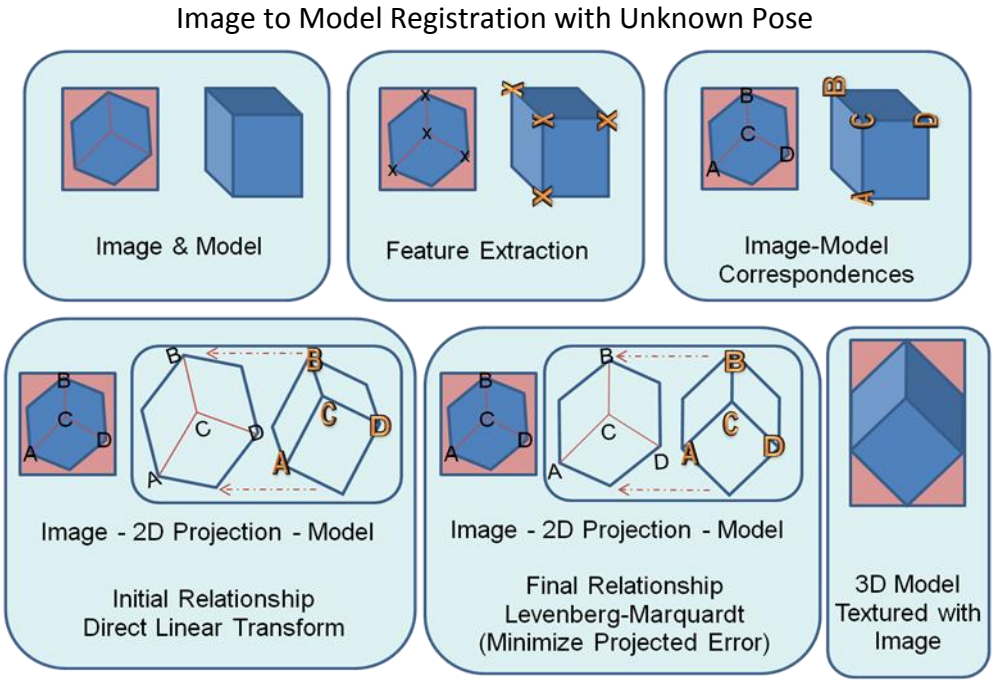


Figure 3-7 The basic process for relating images to a model when the camera pose is unknown. The main difference here is that the initial camera pose must be solved for using correspondences or user manipulation of the model pose. At this point the process then mimics the one described earlier in Section 3.1.

The resectioning approach implemented by the author is similar to the Maximum Likelihood, "Gold Standard Algorithm" proposed by Hartley & Zisserman (Hartley and Zisserman 2004); which is shown in Figure 3-8:

Objective

Given $n \geq 6$ world to image point correspondences $\{\mathbf{X}_i \leftrightarrow \mathbf{x}_i\}$, determine the Maximum Likelihood estimate of the camera projection matrix $\mathtt{P}$, i.e. the $\mathtt{P}$ which minimizes $\sum_i d(\mathbf{x}_i, \mathtt{P}\mathbf{X}_i)^2$.

Algorithm

(i) **Linear solution.** Compute an initial estimate of $\mathtt{P}$ using a linear method such as algorithm 4.2(p109):

  (a) **Normalization:** Use a similarity transformation $\mathtt{T}$ to normalize the image points, and a second similarity transformation $\mathtt{U}$ to normalize the space points. Suppose the normalized image points are $\tilde{\mathbf{x}}_i = \mathtt{T}\mathbf{x}_i$, and the normalized space points are $\tilde{\mathbf{X}}_i = \mathtt{U}\mathbf{X}_i$.

  (b) **DLT:** Form the $2n \times 12$ matrix $\mathtt{A}$ by stacking the equations (7.2) generated by each correspondence $\tilde{\mathbf{X}}_i \leftrightarrow \tilde{\mathbf{x}}_i$. Write $\mathbf{p}$ for the vector containing the entries of the matrix $\tilde{\mathtt{P}}$. A solution of $\mathtt{A}\mathbf{p} = \mathbf{0}$, subject to $\|\mathbf{p}\| = 1$, is obtained from the unit singular vector of $\mathtt{A}$ corresponding to the smallest singular value.

(ii) **Minimize geometric error.** Using the linear estimate as a starting point minimize the geometric error (7.4):

$$\sum_i d(\tilde{\mathbf{x}}_i, \tilde{\mathtt{P}}\tilde{\mathbf{X}}_i)^2$$

over $\tilde{\mathtt{P}}$, using an iterative algorithm such as Levenberg–Marquardt.

(iii) **Denormalization.** The camera matrix for the original (unnormalized) coordinates is obtained from $\tilde{\mathtt{P}}$ as

$$\mathtt{P} = \mathtt{T}^{-1}\tilde{\mathtt{P}}\mathtt{U}.$$

Figure 3-8 Algorithm 7.1 – The Gold Standard Algorithm for estimating P from world to image point correspondences in the case that the world points are very accurately known.

Since the mathematical formulation and execution for this algorithm are treated exhaustively in Chapters 12 & 13, they will only be covered briefly in the following sections to highlight areas of additional interest.

3.2.1 Approach - Feature Extraction and Matching

As noted above, the data that we want to relate in this section is the 2D information from an image to that of a known 3D model. By orienting our model to the viewing geometry of an arbitrary scene image, we wish to determine information about the camera that acquired that imagery of our 3D modeled site. Specifically, we wish to determine the internal and external camera parameters (see Chapter 11).

The case of feature extraction in this scenario is not as straightforward as in the image-to-image registration of Chapter 2. Here, we have a 3D model which we can easily rotate to an orientation that approximates our image view through user assisted computer graphic manipulation. MATLAB's (The Mathworks, Inc. 2010) graphical plotting interface allows these manipulations through simple mouse-driven commands, when the rotation button is active (see figures in the next Case Study - Section 3.2.3).

Once this is accomplished, we can implement a "back-culling" facet routine to extract only those features visible in the Graphical User Interface (GUI) window. In this way, it is possible to isolate all of the vertices that should be present as image corners within a scene. Since faceted models are often overly-simplified renditions of the original scene, these vertices will be a subset of the corners within a scene. For this reason, it is possible to utilize automated techniques to extract the semi-invariant features necessary to match a model with an image utilizing "facet culling" and corner/edge detection techniques.

Once the common invariant features have been extracted from the model and image, the critical and yet daunting task of relating correspondences begins. The problem is challenging in this situation, because of the dissimilarity in the two datasets and uncharacterized error in the model points may make them extremely difficult to relate automatically. Finally, regardless of the specific approach utilized to automatically match features, both the statistical RMSDE and RANSAC techniques can be used for outlier removal and error minimization.

3.2.1.1 *Using the Projected Model and Image*

When a hi-fidelity model is available (which is texture mapped with imagery of the same modality), traditional 2D image registration techniques may be applied to extract correspondences. This was the approach utilized in Section 3.1, for the final registration step. Recall that the model was projected onto a similar 2D plane as the image where features can be extracted via traditional edge detection algorithms such as SIFT (D. G. Lowe 2004), LoGWaR (Walli 2003), or Harris Corner Detector (Harris and Stephens 1988).

If 2D registration of the projected model and image is feasible, direct methods to relate the underlying 3D resectioning relationships between the image and the model using the fundamental matrix (see Section 14.2) and 2D homography can be utilized (Seedahmed 2006). This allows for an automated comparison of the projected model vertices with the extracted invariant match points for potential commonality and precise model to image registration.

3.2.1.2 *Using Feature Distances*

A related method is to relate the image and model features based on a feature distance relationship, such as the technique utilized in LoGWar. Since the user assisted orientation can remove most of the projective effects between the model and the image, it is possible to utilize the projection of the observed vertices for comparison to the extracted image features. The distance relationship is naturally invariant to shift and rotation, and scale invariance is achieved through the ratio of distances. Additionally, the projective effect is invariant when comparing the cross ratio of distance ratios along a line. As long as the dissimilarity in point sets doesn't preclude a robust solution, this technique can also provide a viable automated solution.

3.2.1.3 *Using Semi-Automated Tools and User Assistance*

If completely automated techniques prove to be too difficult to provide a linear estimate of the image resectioning solution, manual selection of no less than 6 model vertices that are visible within the image may be required. Of course, due to error in the model representation of the real world, image point selection and other error, additional correspondences will provide increased accuracy when using least squares estimated solutions (Section 3.2.3).

3.2.2 Develop Linear and Non-Linear Solutions

An initial linear estimate of the camera pose is obtained by performing a Direct Linear Transform (DLT). A more detailed discussion regarding the DLT's implementation w.r.t. estimating the full 11 parameter camera pose problem is available in Appendix B (Chapter 12). Once an initial linear estimate of the solution "puts us in the ball park" of the correct solution, it can often provide nonlinear techniques with a faster solution that has greater likelihood of converging at the true global minimum. A simple graphic that can help visualize this concept is shown in Figure 3-9.

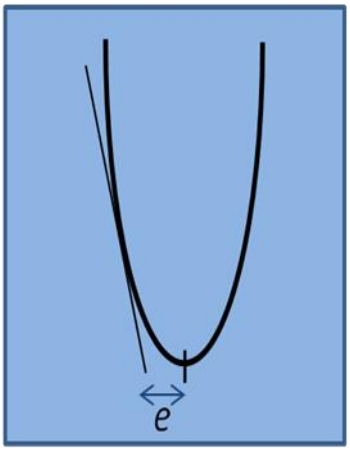


Figure 3-9 This simple graphic displays how a linear estimate of a nonlinear function can provide a rough estimate of the local/global minimum location, within some margin of error.

Also, it is often possible to simplify the solution space by making initial assumptions. For example, in near-orthographic imaging scenarios, it is often acceptable to assume the camera roll and pitch are negligible (~0^{o}), for initial estimation purposes (DeWitt and Wolf 2000). Additionally, a user could easily obtain an initial estimate of the camera parameters by rotating, scaling, and translating a model to the approximate orientation and position displayed in the image.

Unfortunately, due to the inherently nonlinear interactions of the camera pose parameters, a linear solution will normally be insufficient to provide the required accuracies necessary to relate a 3D model with an image. In this case, an iterative nonlinear estimation process will normally be required to arrive at satisfactory results. Due to the proven performance of the Levenberg-Marquardt Algorithm (LMA) to efficiently and robustly solve for many nonlinear problems (Hartley and Zisserman 2004), we will utilize it here to solve for the camera pose parameters.

The LMA is a hybrid of the Gauss-Newton algorithm (GNA) and the method of gradient decent. Although it tends to be more robust than GNA when starting far from the minimum, it often converges more slowly to that minimum (H. Rhody 2009). Additional details regarding the general implementation of the LMA with specific application to the resectioning of images-to-models is provided as a reference in Chapter 13.

As in any iterative solution, the key metric to adequately quantify is that which you are minimizing against. In the case of 2D imagery features and 3D model control points, that metric is the geometric distance between the 2D features and the projected 3D model control points into that 2D space.

Minimization Equation

$$min \sum_i d(x_i, PX_i)^2 \quad \textbf{(40)}$$

Where P is the projection matrix, PX_i is the location of the projected 3D model point onto the 2D image space and x_i is the corresponding image feature. The solution is the minimum of the total (summed) square error over all the points considered. If the measurement errors are Gaussian, this will be the Maximum Likelihood estimate of P (Hartley and Zisserman 2004).

Specific to the LMA, a damping factor μ is applied that weights the direction and step size of the decent into the minimization valley. When large, this damping term delivers a short step in the steepest descent direction; which is good if the current iteration is far from the solution. However, when μ is small, it is possible to achieve nearly quadratic final convergence (Madsen, Nielsen and Tingleff 2004).

3.2.3 Case Study - Estimating Model Pose from unknown Imagery

This example demonstrates the ability to recover an unknown camera's pose of a scene, when it acquired an image of the Center for Imaging Science (CIS) at RIT. In this study, the model vertices and corresponding image locations (12 GCPs) were selected manually and are visible in Figure 3-10.

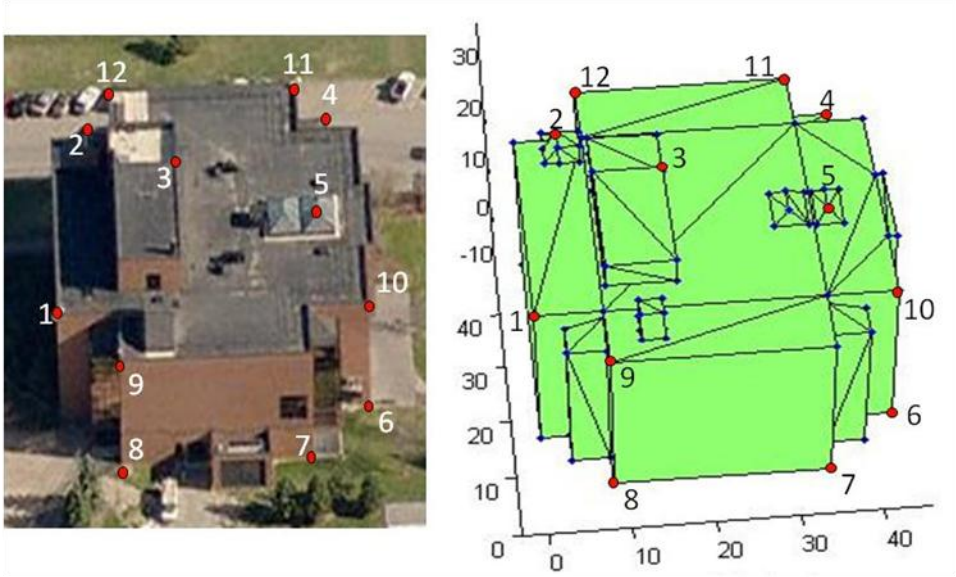


Figure 3-10 On the left is the working image with the same 12 locations selected as on the model; these locations are twice the number required for resectioning with a model (6 GCPs).

In its attempt to minimize error, LMA took the initial sum of the squared projected geometric error from 16 million [pix] to 25 [pix], after only 29 iterations. Not only did the total squared error reduce drastically, the parameter minimization provided good results, which are visible in Figure 3-11. Here the resectioning is utilized to determine the DLT estimate on the left and the LMA optimization on the right.

DLT Estimate – 12pts DLT then LMA – 12pts

Figure 3-11 On the left, the DLT provides a good starting point for LMA to optimize a solution.

This study demonstrates the ability to determine a modeled scene's pose w.r.t. an image. Future research in this area will attempt to develop automated techniques to relate the model and the working image. Key to enabling this would be to automate the matching of model vertices to corresponding image locations. This could be accomplished by projecting the vertices onto the image plane and matching them with HCD features, using the distance matching technique covered in Section 2.2.1.

Another approach is to match the projected model image to the working image, as demonstrated in the previous case study. This requires accurate modeling of the scene, but, allows the use of traditional 2D image registration technique to derive correspondences. The point correspondences can then be utilized to solve for the resectioning parameters directly from a 2D projective transformation (Seedahmed 2006).

Additionally, the DLT and LMA techniques covered in Chapter 12 & 13 can be applied to image-to-image resectioning, instead of image-to-model. Finally, the estimated structure of the scene can be constrained by the known model facets to limit the projective ambiguity common in image sparse structure reconstructions.

3.3 SWIR Imagery to SWIR Attributed LIDAR Models

In this section we will explore an interesting example of completely automated 2D imagery to 3D model registration utilizing WASP imagery and LIDAR data (Kucera International Inc. 2010), as seen in Figure 3-12.

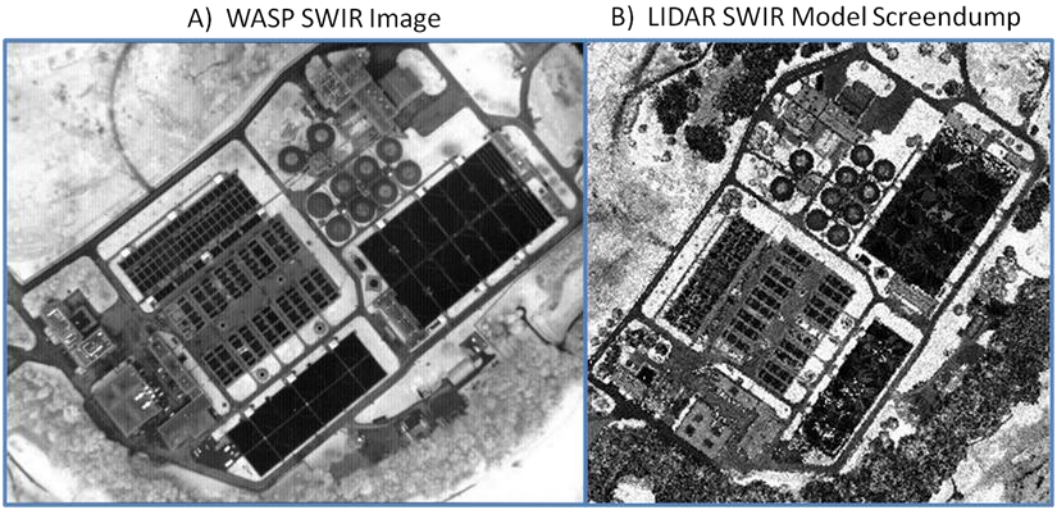


Figure 3-12 The figure above show a 2D SWIR image (A) and an image projection of a 3D model that was textured/attributed using the same LIDAR SWIR intensity returns that were utilized to create the facetized 3D model.

Below, in Figure 3-13, the two datasets were related automatically using techniques developed in Chapter 2. It is useful to note that, in this example, no sensor viewing orientation was utilized to estimate the sensor-to-scene view in order to remove the 3D projective effects and a good registration was still possible.

Automated Image-to-Model registration using SIFT & RANSAC.

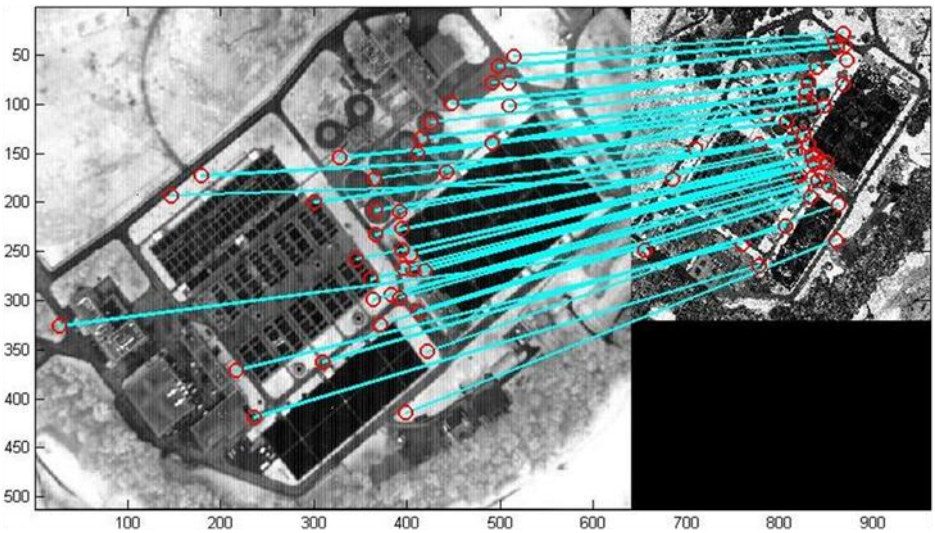


Figure 3-13 The results of automated registration (using SIFT & RANSAC), between the 2D SWIR image and the 3D LIDAR model are apparent.

The robustly matched correspondences can be utilized to relate via a 3D Homography in-order to re-orient the model view (using linear or non-linear techniques) until minimal error exists with the imaged view of the scene (Section 3.2). Alternatively, if a good 2D planar relationship can be derived from the matched correspondences, as discussed in Section 3.1.2, the EOPs can be directly recovered and used to reorient the LIDAR model (Seedahmed 2006). Once corrected, the model orientation should be suitable for direct archival of the imagery onto the model via projective texturing.

In the next section we will leverage some of the techniques developed here, since Chapter 4 focuses on the task of deriving coarse 3D models from only 2D imagery. So it should be apparent that the 3D mathematical projections of models into a related 2D space for registration provide a critically important mathematical framework for this section.

4 Deriving Sparse Structure from Images

Using multiple view imagery to derive sparse structure is known in the photogrammetry community as Bundle Adjustment (BA) and in the computer vision community as Structure from Motion (SfM). Since the BA technique has been around for decades, why is there such a current "Buzz" in scientific literature about its application and utility?

This area of research has recently experienced a renaissance, due to its successful application to several computer/robotic vision projects. The quest to have robots perceive their surroundings with some degree of 3D knowledge, cheaply and robustly, has innumerable applications. To accomplish this feat, the computer vision community has turned to inexpensive cameras and dusted off the photogrammetrist's technique of BA. However, to make robots react to an ever changing environment, they needed to "speed up" the enormously unwieldy BA implementation. To do this efficiently requires sparse matrix techniques, thus the name Sparse Bundle Adjustment (SBA).

Additionally, the mathematical formalism provided by Hartley and Zisserman's Multiple View Geometry (Hartley and Zisserman 2004) text has provided a much needed foundation in this quickly developing area. Finally, for a robot to "see", it must be able to efficiently and robustly extract invariant features from its surroundings via the 2D imagery it has as its source of perception. With proven feature detection algorithms like SIFT, this now becomes feasible. But, it is only the parallel breakthroughs in these areas are finally allowing the dream of rudimentary computer vision to be fulfilled. It is fitting that the remote sensing community

benefit from this as well, especially since the seeds of computer vision were planted over a generation ago by early photogrammetrists. The basic process for recovering 3D structure from images is depicted in Figure 4-1.

Recovering 3D Structure from multiple 2D Images (3D from n2D).

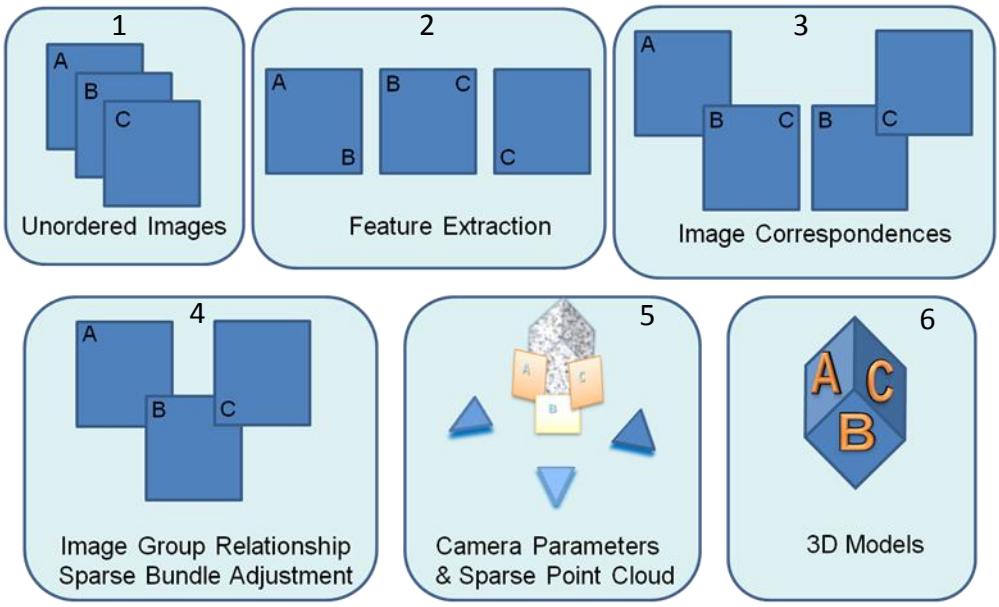


Figure 4-1 This graphic depicts the six basic steps required for relating multiple images to recover sparse structure via the Bundle Adjustment process. Once invariant features are extracted and matched, a linear estimate of the 3D point set is fed into a Bundle Adjustment process to simultaneously optimize the model points and camera parameters.

Two processes for recovering sparse structure will be covered in the following sections. Both techniques utilize SIFT and SBA to "bookend" the beginning and end of the structure recovery process. The first method is a combination of epipolar constraints combined with proven photogrammetric tools, such as the collinearity equation and image rectification to deliver world coordinate system structure from known camera parameters. The second process is entirely enabled by multiple view epipolar geometry methods and provides relative local structure recovery even when most of the camera parameters are unknown.

4.1 Feature Extraction and Matching

Since both of these approaches use the same technique for image feature extraction and matching, it will be covered separately to avoid redundancy. It should be noted that the invariant features used to relate the images for SfM processing are the same features for which the 3D structure is computed and compose the resulting Sparse Point Cloud (SPC).

As in Chapter 2, invariant feature are extracted from the images of overlapping content. These features then need to be matched for potential correspondence. Here we will utilize the SIFT algorithm and its extracted keypoints to match the description vectors that are the closest and assign them as potential matches in each pair of images. Once this is done, the image sets are tested for the requisite number of matches, determined by the number of correspondences necessary to solve for the Fundamental Matrix (7-8 points plus outlier probability) and/or the 2D Homography (4 points plus outlier probability).

The following diagram, adapted from (Hartley and Zisserman 2004), helps depict this epipolar constraint (Figure 4-2). In this diagram the Fundamental Matrix F, dictates that for a given model point X on plane π, a ray must pass from the camera center C (a focal length behind the image plane) through the image location x and this ray will be imaged by the camera C' as an epipolar line l', passing from the image of the same model point x' to that cameras epipole e'. The epipole is the image of the other camera center (which may be off the image). Thus,

$$Fx = l' \qquad \textbf{(41)}$$

and so, $x'^T F$ must be in the left null-space of x and Fx must be in the right null-space of x'^T.

$$x'^{T} F x = 0 \tag{42}$$

So, for a given point x, the preliminary match point must lie along the epipolar line l' in order for it to be valid. So, the matches that do not fit this epipolar constraint are then culled.

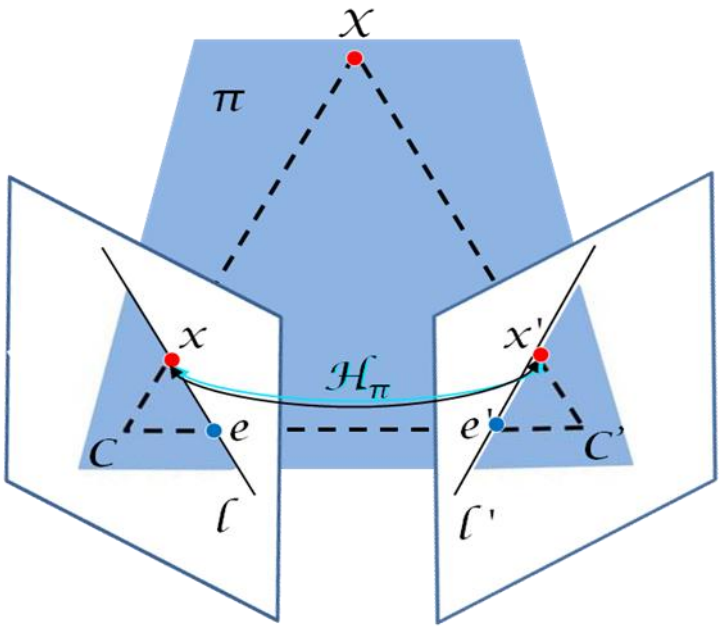


Figure 4-2 The epipolar relationships of the cameras, image points, and model points.

Even in the presence of several outliers, these relationships can be utilized in concert with RANSAC (Section 2.5.2) to develop a robust Fundamental Matrix. Once this is accomplished, the Fundamental Matrix can then be used to constrain the SIFT match set to remove most outliers. These steps can be accomplished using the '*7-Point Fundamental Matrix*' algorithm (Hartley and Zisserman 2004) depicted in Figure 4-3. Unfortunately, it is possible that an erroneous set of correspondences may still fulfill the Fundamental Matrix constraints, so additional constraints may be required to further cull the data (4.2.1).

Objective Compute the fundamental matrix between two images.

Algorithm

(i) **Interest points:** Compute interest points in each image.
(ii) **Putative correspondences:** Compute a set of interest point matches based on proximity and similarity of their intensity neighbourhood.
(iii) **RANSAC robust estimation:** Repeat for N samples, where N is determined adaptively as in algorithm 4.5(p121):
  (a) Select a random sample of 7 correspondences and compute the fundamental matrix F as described in section 11.1.2. There will be one or three real solutions.
  (b) Calculate the distance $d_\perp$ for each putative correspondence.
  (c) Compute the number of inliers consistent with F by the number of correspondences for which $d_\perp < t$ pixels.
  (d) If there are three real solutions for F the number of inliers is computed for each solution, and the solution with most inliers retained.

  Choose the F with the largest number of inliers. In the case of ties choose the solution that has the lowest standard deviation of inliers.
(iv) **Non-linear estimation:** re-estimate F from all correspondences classified as inliers by minimizing a cost function, e.g. (11.6), using the Levenberg–Marquardt algorithm of section A6.2(p600).
(v) **Guided matching:** Further interest point correspondences are now determined using the estimated F to define a search strip about the epipolar line.

The last two steps can be iterated until the number of correspondences is stable.

Figure 4-3 Hartley & Zisserman's 7-Point Fundamental Matrix using RANSAC.

4.2 Modern Photogrammetric Techniques

Automated synthetic scene generation is now becoming feasible with calibrated camera remote sensing. This section implements computer vision techniques that have recently become popular to extract "structure from motion" (SfM) of a calibrated camera with respect to a target. This process is similar to Microsoft's popular "PhotoSynth" technique (Microsoft, 2009), but, blends photogrammetric with computer vision techniques and applies it to geographic scenes imaged from an airborne platform. Additionally, it has been augmented with new features to increase the fidelity of the 3D structure for realistic scene modeling. This includes the generation of both sparse and dense point clouds useful for synthetic macro/micro-scene reconstruction.

Although, the quest for computer vision has been an active area of research for decades, it has recently experienced a renaissance due to a few significant breakthroughs. This section will

review the developments in mathematical formalism, robust automated point extraction, and efficient sparse matrix algorithm implementation that have fomented the capability to retrieve 3D structure from multiple aerial images of the same target and apply it to geographical scene modeling.

Scenes are reconstructed on both a macro and a micro scale. The macro scene reconstruction implements the scale invariant feature transform to establish initial correspondences, then extracts a scene coordinate estimate using photogrammetric techniques. The estimates along with calibrated camera information are fed through a sparse bundle adjustment to extract refined scene coordinates. The micro scale reconstruction uses a denser correspondence done on specific targets using the epipolar geometry derived in the macro method.

4.2.1 Approach – Depth Recovery from Overlapping Images

The basic method for implementing the Modern Photogrammetric approach is to:

1. *Derive Initial Correspondences utilizing the SIFT Algorithm*
2. *Cull Outlier Matches for Precise Image Bundle Relationship and 3D Structure*
    a. *Check for agreement with the Fundamental Matrix using RANSAC*
    b. *Check for general agreement with a planar SPC fit using RANSAC*
3. *Rectify Images by projecting points onto a virtual focal plane*
4. *Estimate the 3D structure utilizing linear techniques*
5. *Determine Correspondences with multiple image matches*
6. *Prepare Match Datasets for Sparse Bundle Adjustment (SBA)*
7. *Relate SBA results to WCS using Camera info and Back Projection*

4.2.1.1 *Derive Correspondences*

Deriving correspondences is implemented similar to Section 2.2.2, but, here the author has implemented an image tiling approach that overcomes the self-imposed 2k x 2k limitation of Dr Lowe's SIFT implementation currently available from his website (D. Lowe 2005). This technique provides about 5x the number of invariant features and 3x the correspondences as the reduced resolution imagery and is implemented as shown in Figure 4-4. Here the Hi-Resolution and Low-Resolution tiles are layered to provide the proper combination for multiscale image pyramid analysis within the SIFT algorithm.

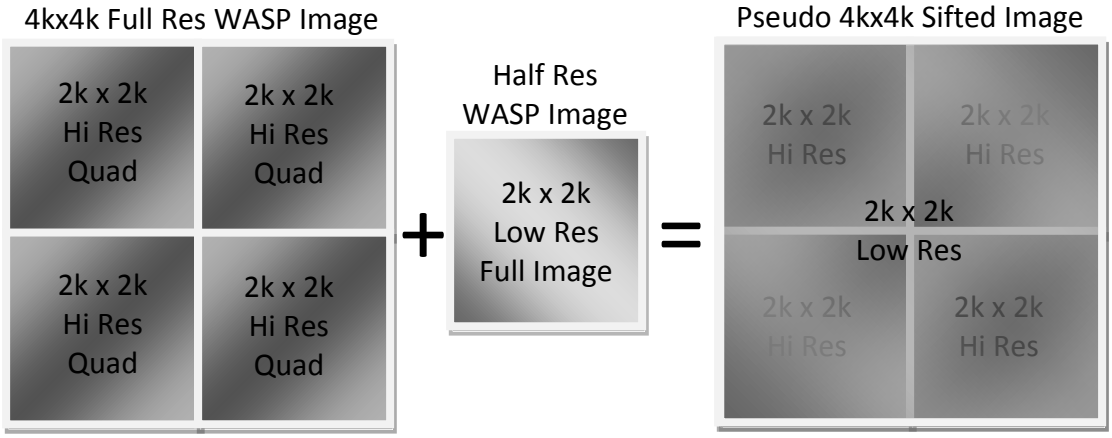


Figure 4-4 Process for tiling images larger than 2kx2k for SIFT feature extraction and matching.

4.2.1.2 *Culling Outliers*

First the SIFT correspondences are run through a RANSAC algorithm constrained against the resulting Fundamental Matrix as in Section 2.5.2. This compares the candidate feature matches against the epipolar relationships derived from the initial point set. Matches that do not support this relationship (42) are culled as outliers. Occasionally outliers may still fulfill this requirement and additional techniques for outlier removal are required.

Thankfully, additional culling of outlier matches can be applied to most remotely sensed image bundles, due to the near-planar target terrain. Once the linear estimate of the elevation of each point in the match set has been obtained, a RANSAC plane fitting technique can often be utilized to remove any remaining outliers (i.e. $\pm 1\, Std$ of the distribution from the plane or $\pm 30m$) as demonstrated in Figure 4-5.

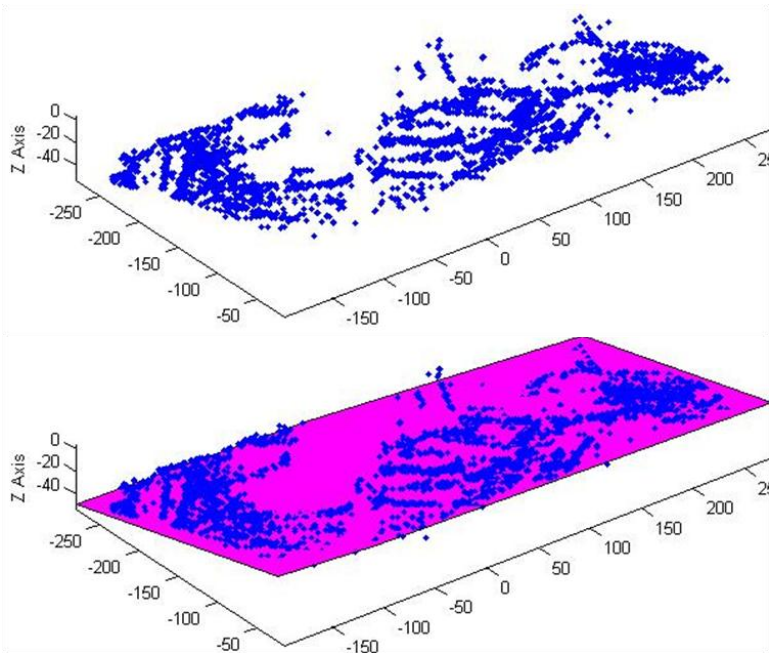


Figure 4-5 Displays the utility of RANSAC plane fitting to SPC terrain data for outlier removal.

4.2.1.3 *Rectify Images using the Collinearity Equation*

Unless the acquisition platform is accomplishing purely nadir imaging (looking perpendicular to the earth's surface) it is necessary to rectify the image or image correspondences to enable proper linear 3D structure estimation. The approach taken here is to back-project the image correspondences onto a virtual focal plane that is located at the focal length (f), but, is situated perpendicular to the earth's surface as depicted in Figure 4-6. This is a critical correction that generalizes the linear 3D recovery techniques covered in Section 4.3.2.4.

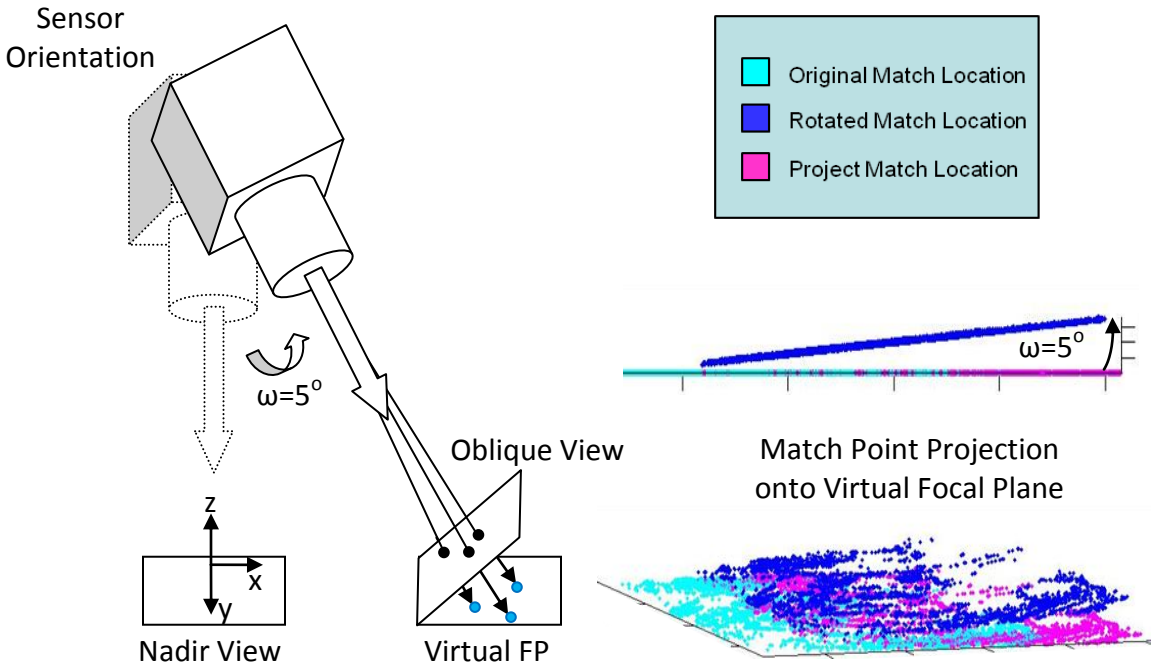


Figure 4-6 Rectification of the matches must be performed for accurate 3D estimation of the SPC.

4.2.1.4 *Estimate the 3D Structure – Linear Photogrammetric*

The basic technique to derive 3D structure from images was derived from a photogrammetric approach (DeWitt and Wolf 2000). Although this technique is easy to comprehend and implement, it has severe limitations for use. The reason for this is that it assumes the sensing platform is performing Nadir Imaging along a flight path that is parallel to one of the image axes. This essentially means that there is no pitch and roll and the heading is constant w.r.t. the other images and runs in straight lines. Unfortunately, with airborne platforms this is seldom the case and corrections must be incorporated for robust performance. The previous section corrected for the pitch and roll of the sensor, but, we still must accommodate for the deviation of the image axes from the flight line. This is covered in Section 4.3.2.4 and visible in Figure 4-7.

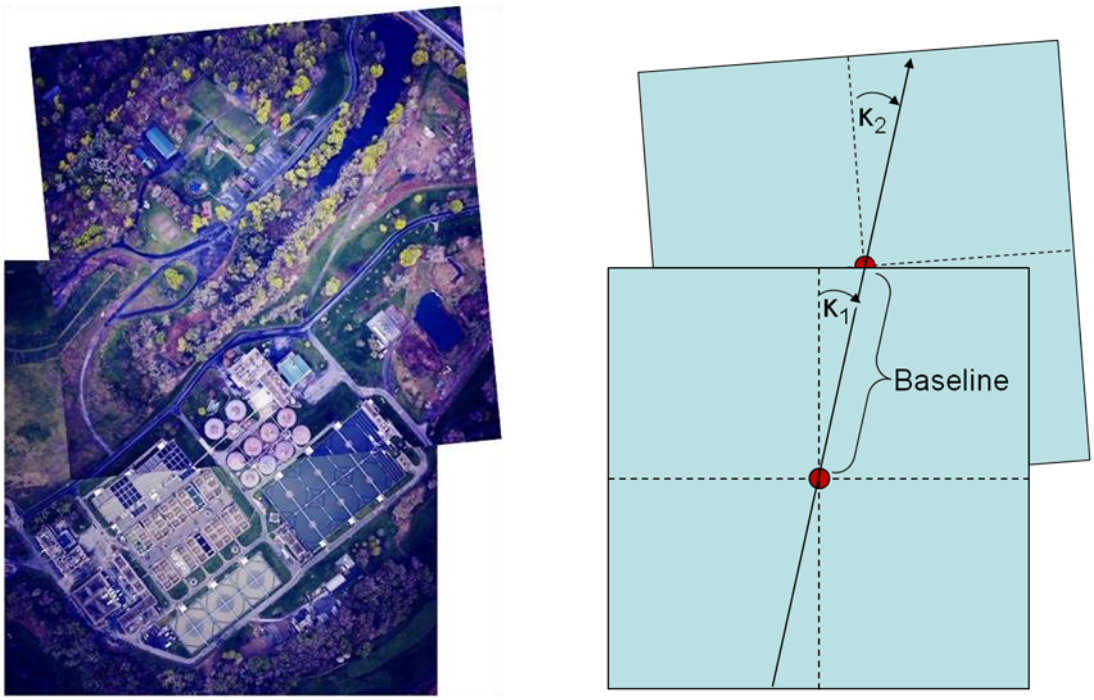


Figure 4-7 The 3D estimate of structure is dependent on the baseline between the images, so corrections are required that change the image pixel locations to be aligned with the flight line path. This amounts to a coordinate system conversion of the matched locations to one that is defined by the axes connecting both camera location at the time of acquisition.

4.2.1.5 *Isolate Correspondences with Multiple Matches and Prepare for SBA*

Once the 3D estimate of the matches is accomplished, the next step is to group the image matches into sets, based on location similarity as displayed in Figure 4-8. It should be no surprise that similar regions are isolated by the SIFT algorithm across multiple images, due to the gradient nature of the feature mapping. In this way, a set of images may have a few features that are isolated in every image that has common overlap within the set.

The SBA algorithm of Lourakis and Argyros (Lourakis & Argyros 2004) is optimized for speed and efficiency and is utilized in Section 4.4 to provide an optimized point cloud w.r.t the camera's EOPs/IOPs and 3D point locations. It can easily optimize against several camera variables and the structure of tens of thousands of 3D points simultaneously to produces an image bundle

that is mutually self-consistent. However, as with any engineering code, it requires specific formatting for the input variables and special care when preparing the camera IOPs and EOPs.

The SBA code weights the matches based on the number of correspondences and their projected covariance. For this reason, a sorting algorithm has been developed and implemented by the author to extract match sets from the image bundle in a pyramid fashion, where the 4 match set is extracted from the 3 match set which is in turn extracted from the most common 2 match set to remove redundancy before sending them into the SBA algorithm.

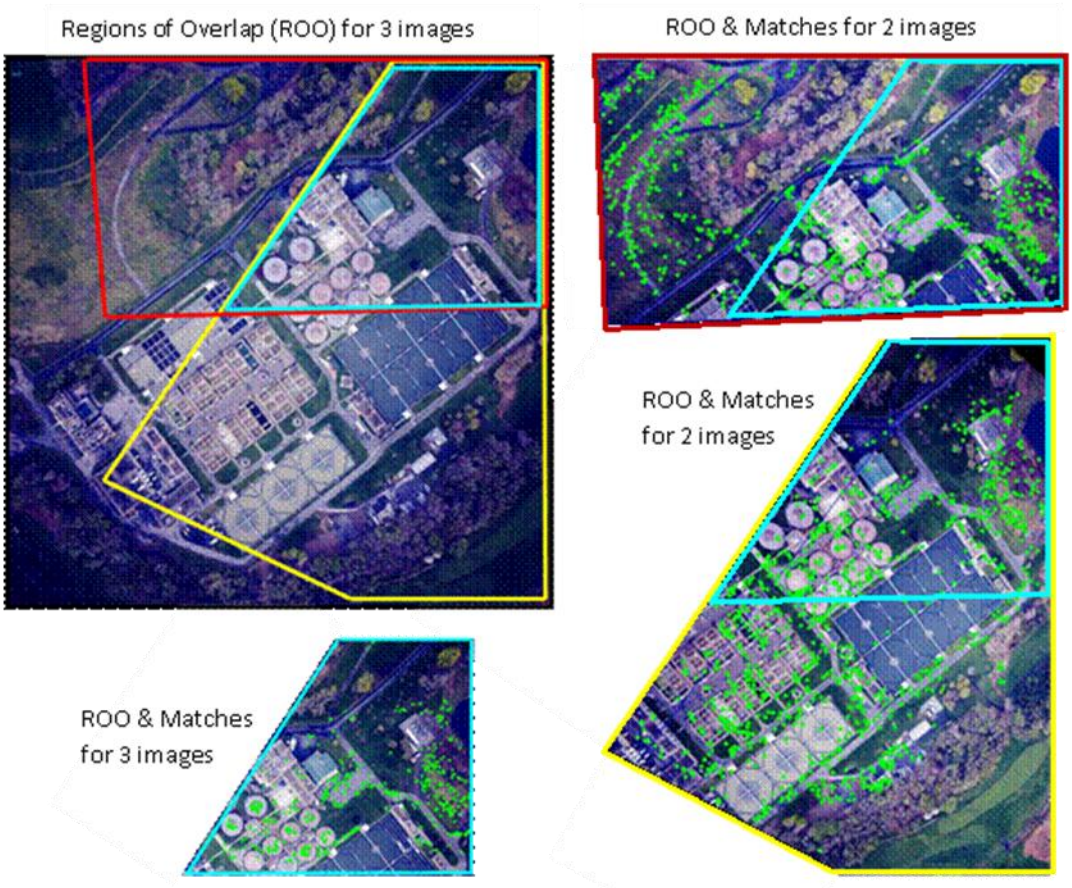


Figure 4-8 The overlapping images above (red & yellow) are registered and have matches that are common to all (cyan). These common locations can then be utilized for 3D registration or as seeds for the DPC extraction process (Section 4.3.3).

4.2.1.6 *Relate SBA Results to WCS using the Collinearity Equation*

Once a good image bundle and SPC is produced using SBA, it can be related back to the WCS directly by utilizing the camera Exterior Orientation Parameters (EOPs) and Interior Orientation Parameters (IOPs) which can also be optimized in this process. Additionally, the recovered height information can be utilized in concert with the Collinearity Equations to re-project into any given image orientation as shown below in Figure 4-9.

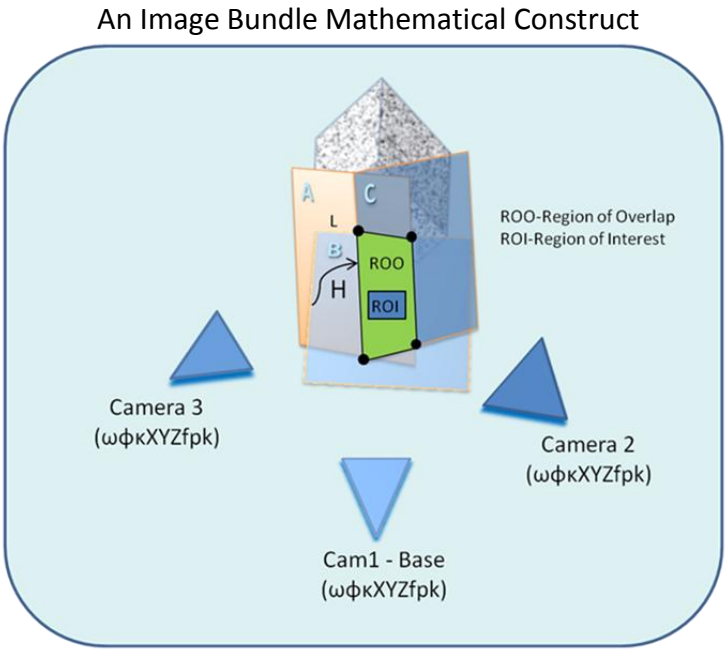


Figure 4-9 Once the image bundle is optimized using SBA, it is possible to relate the images, cameras and 3D point cloud into a 3D mathematical framework to determine the region of overlap for DPC interrogation and additional processing.

The last step in the image bundle process is to automatically identify the Regions of Overlap (ROO) of the resulting image relationships (Figure 4-9) for seamless integration of Dense Point Cloud (DPC) processing (Nilosek, et al. 2009). This process utilizes the Fundamental Matrix relationship between two matched images and interrogates each pixel of the base image to develop a correspondence in the working image as shown in Figure 4-10. Since we know that

for any given pixel in the base image there must be a corresponding pixel (if not occluded) located in the working image, it is efficient to look along the epipolar line for a match to constrain the search. Due to intense interest by the computer vision community, active research is currently ongoing in this field with some promising initial results (Pollefeys, et al. 2004) & (Ma, et al. 2006), but, still with much room for growth and discovery.

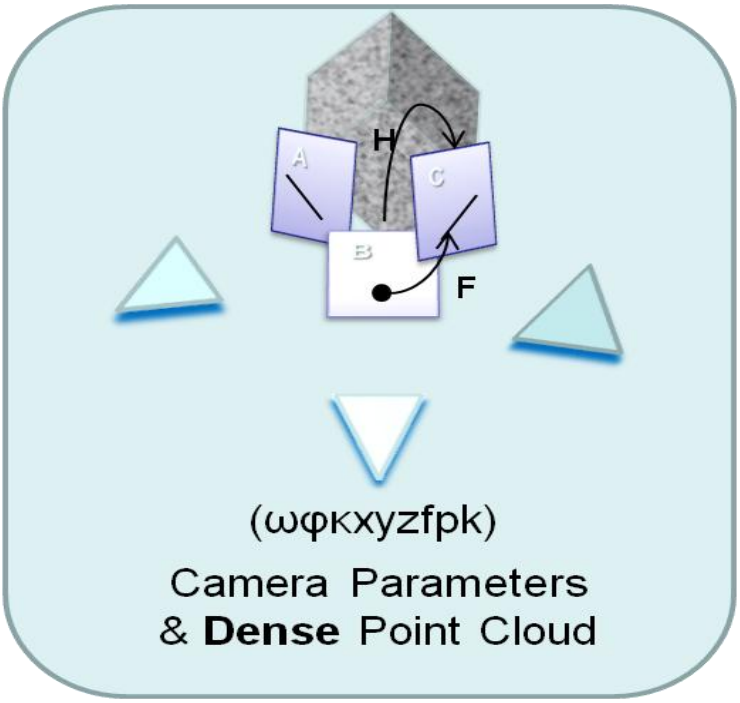


Figure 4-10 The basic process for developing Dense Point Clouds using Epipolar relationships between images.

Even though the epipolar constraint for deriving dense correspondences greatly reduces the search space, the effects of parallax and occlusion may greatly change the localized region's appearance. However, there are some automated techniques like the Affine SIFT (ASIFT) (Morel & Yu, 2009) or the Log Polar algorithm (Cyganek 2008), that could be utilized to provide

scale, shift, and rotation invariant approaches for dense correspondence matching and outlier removal to address this challenging feature matching problem (Nilosek, et al. 2009).

4.3 Case Study – Creating Sparse Structure using Airborne Data

The seeds of computer vision were actually planted by photogrammetrists over 40 years ago, through the development of "space resectioning" and "bundle adjustment" techniques. But it is only the parallel breakthroughs, in the previously mentioned areas that have finally allowed the dream of rudimentary computer vision to be fulfilled in an efficient and robust fashion. Both areas will benefit from the application of these advancements to geographical synthetic scene modeling. This section explores the process the authors refer to as Airborne Synthetic Scene Generation (AeroSynth) process (Walli, Nilosek, et al. 2009).

The AeroSynth technique for recovering 3D structure from images is a blend of the both the photogrammetric and computer vision approaches. It utilizes the automatic feature isolation/matching, epipolar relationships and SBA of computer vision and melds it with the linear 3D point estimation and collinearity relationships of photogrammetry. As a result, the image bundle and SBA-SPC can be related to the WCS and directly injected into GIS applications for automatic analysis and comparison to existing archival data.

4.3.1 AeroSynth Introduction

Recovering 3D structure from 2D images requires only that the scene is imaged from two different viewing geometries and that the same features can be accurately identified. Figure 4-11, depicts a site of interest imaged from multiple views using an airborne sensor; here the

point of interest is the top of a smokestack that will be imaged with the effects of parallax displacing it with respect to other features within the scene. This parallax displacement effect has been used for decades within the photogrammetry community to recover the 3D structure within a scene (DeWitt and Wolf 2000). Unfortunately, robust automated techniques to match similar features within a scene have been fairly elusive until very recent breakthroughs in the area of computer vision (Section 2.2).

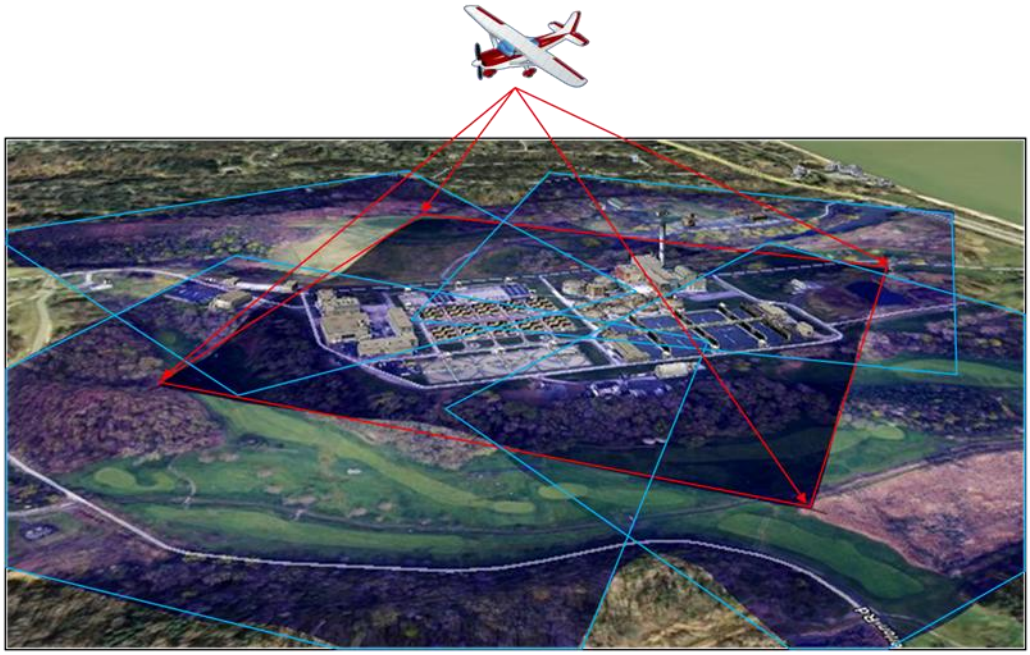


Figure 4-11 Example showing the angular diversity required to recover 3D Terrain from Airborne Imagery.

4.3.2 Recovering Sparse Structure from Images

The key to automatically recovering 3D structure from an imaged scene is to identify reliable invariant features, match these features from images with diverse angular views of the target and then generate accurate mathematical relationships to relate the images. This information can then be utilized in concert with the camera external and internal orientation parameters to derive scene structure that is defined within the World Coordinate System (WCS) of choice.

4.3.2.1 *Airborne Dataset*

For this study, the working imagery was obtained from the Rochester Institute of Technology, Center for Imaging Science's (RIT/CIS), Wildfire Airborne Sensing Program (WASP) multimodal sensor suite (Rhody, Van Aardt, Faulring, & McKeown, 2008). This sensor provides 4kx4k Visible Near-Infrared (VNIR) and 640x512 Shortwave Infrared (SWIR), Midwave Infrared (MWIR), and Longwave Infrared (LWIR) images. Google Earth (GE) was utilized as the GIS visualization tool, with a detailed model of the Frank E. VanLare Water Treatment Plant (Pictometry, 2008) embedded within the standard satellite imagery and 30 [m] terrain elevation maps (Figure 4-11 & Figure 4-14). Additionally, Figure 4-11 shows the region of overlap (outlined in red) of 5 WASP images where the site of interest is contained in the central (base) image.

4.3.2.2 *Invariant Feature Detection and Matching*

The SIFT technique can consistently isolate thousands of potential invariant features within an arbitrary image as seen in Figure 4-12. This is extremely useful when attempting to create sparse structure from matched point correspondences, since any matching features can then be processed to obtain the 3D structure of the imaged scene. In addition, more recent independent testing has confirmed that the SIFT feature detector, and its variants, perform better under varying image conditions than other current feature extraction techniques (Moreels and Perona 2006) & (Mikolajczyk and Schmid 2005)

The SIFT algorithm utilizes a Difference of Gaussian edge detector of varying widths to isolate features and define a gradient mapping around them. These gradient maps are then compared for similarity in another image and matches result from the most likely invariant feature pairs.

Once potential matches are found, outliers can be culled based on the requisite epipolar relationships that must exist between two images of the same scene. This has always been challenging in the past due to the effects of parallax, but, can now be robustly addressed using techniques highlighted in the next section.

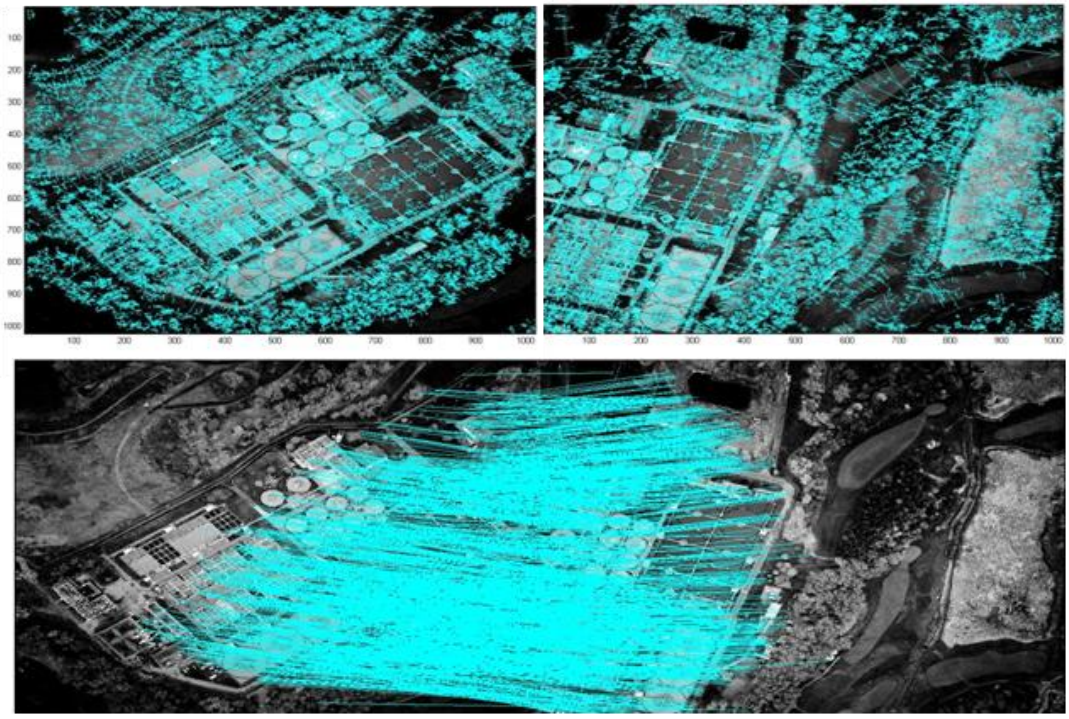


Figure 4-12 Thousands of invariant keypoints generated and matched using the SIFT algorithm.

4.3.2.3 *Outlier Removal*

To successfully remove erroneous matches derived using the SIFT algorithm, the potential match set will be processed using the Random Sample Consensus (RANSAC) technique (Fischler and Bolles 1981), in conjunction with the fundamental matrix relationship between images of the same scene (Figure 4-12). RANSAC has proven to be a robust technique for outlier removal, even in the presence of large numbers of incorrect matches (Hartley & Zisserman, 2004). Since it is not necessary to test all the sets of points for a solution, it can be efficiently utilized with techniques like SIFT that provide large numbers of automated matches.

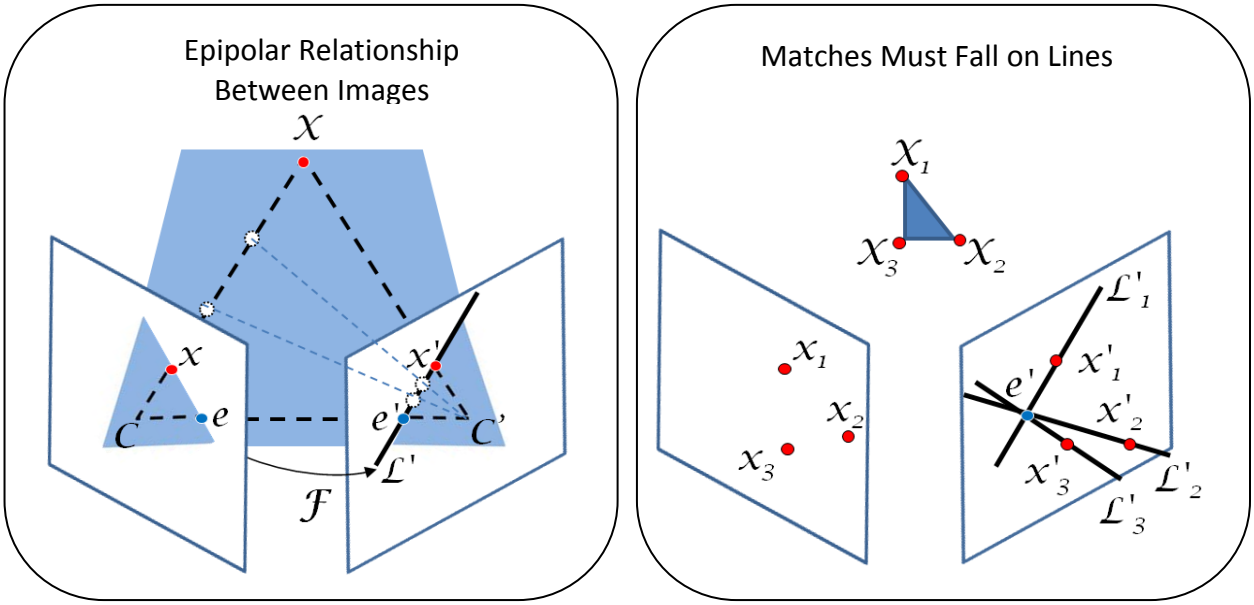


Figure 4-13 Depiction of the Fundamental Matrix constraint between images which is used for outlier removal.

In the diagram above, Figure 4-13, the Fundamental Matrix ***F*** dictates that for a given 3D scene point ***X***, a ray must pass from the camera center ***C*** (a focal length behind the image plane) through the image location ***x*** and this ray will be imaged by the camera ***C'*** as an epipolar line ***l'***, passing from the image of the same model point ***x'*** to that cameras epipole ***e'*** (Hartley and Zisserman 2004). The epipole is the image of the other camera center (which may be off the image entirely).

Anyone that has worked for any length of time with automatic image registration can attest to the challenging issues parallax can cause when relating features. The limitation of utilizing a 2D Projective Homography to relate imagery with large elevation difference between acquisition stations, can be addressed through the use of the Fundamental Matrix relationship. This relationship constrains the matches to an epipolar line even under extreme parallax situations and can be formalized in a mathematical manner as shown below (Hartley and Zisserman 2004).

Fundamental Matrix

$$Fx = l' \tag{43}$$

So, $\boldsymbol{x'^{T}F}$ must be in the left null-space of $\boldsymbol{x}$ and $\boldsymbol{Fx}$ must be in the right null-space of $\boldsymbol{x'^{T}}$.

Fundamental Null Space

$$x'^{T}Fx = 0 \tag{44}$$

Simply stated, for a given point $\boldsymbol{x}$, the preliminary match point must lie along the epipolar line $\boldsymbol{l'}$ in order for it to be a valid match. So, the proposed feature matches that do not fit this epipolar constraint are considered bad matches.

Once the initial matched point set has been obtained using the automated SIFT technique, it is usually necessary to test for these bad matches or "outliers". The RANSAC algorithm can be utilized to iteratively take a random sample of the matches to create a Fundamental Matrix relationship between the images. Once this is done, the veracity of that relationship can be tested by comparing the number of resulting inliers against a statistically relevant number of additional tests. The Fundamental Matrix that produces the most match point inliers is then accepted as the best mathematical model and any outliers to this model are then removed. These procedures are detailed in Section 2.5.2 and can produce thousands of good matches, per image pair. These 2D image correspondences are then processed into 3D points using the linear techniques described in the next section.

4.3.2.4 *Initial Estimate of Sparse Structure*

The initial estimation technique that is utilized to derive the 3D scene structure utilizes a simple

approach that is augmented for more general situations by compensating for the aircraft motion and image axes misalignment with the flight path. This process can be visualized below in Figure 4-14.

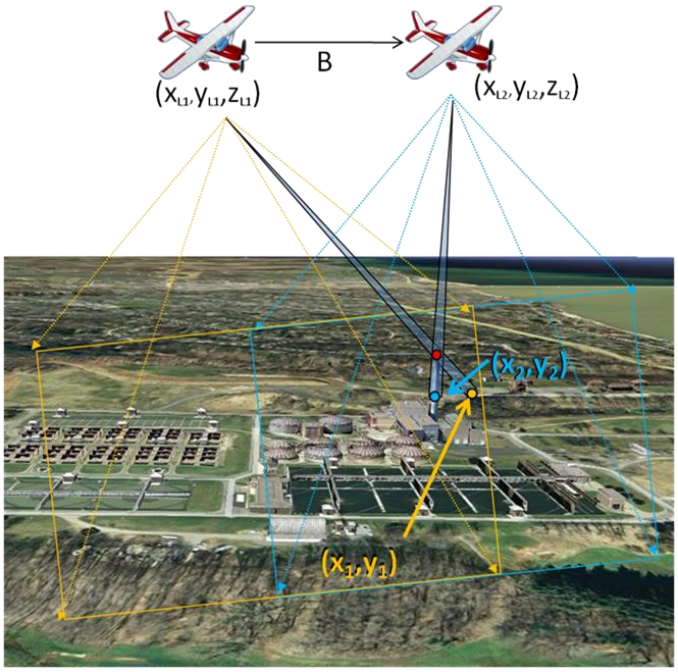


Figure 4-14 Graphic showing two collection stations of an airborne sensor utilized to recover 3D Structure.

The following equations (DeWitt and Wolf 2000) can then be utilized to derive 3D structure once the necessary corrections have been accomplished. Here C_{xi} and C_{yi} are the longitude and latitude of the cameras and C_{z1} is the flying height of the base sensor, B is the baseline distance between sensor locations (the airbase), p_i is the pixel distance between matching points (the distance here is only along the x-axis), and the pixel locations are denoted (x_{1i}, y_{1i}) and (x_{2i}, y_{2i}).

Baseline Distance (x-axis)

$$B = C_{x2} - C_{x1} \quad \textbf{(45)}$$

Focal Plane Distance (x-axis)

$$p_i = p_{xi} = x_{1i} - x_{2i} \quad \textbf{(46)}$$

WCS Relative Longitude

$$X_i = \frac{B * x_{1i}}{p_i} \quad \textbf{(47)}$$

WCS Relative Latitude

$$Y_i = \frac{B * y_{1i}}{p_i} \quad \textbf{(48)}$$

WCS Relative Altitude

$$Z_i = C_{z1} - \frac{B * f}{p_i} \quad \textbf{(49)}$$

Figure 4-15 depicts the corrections that are required for any deviation of the flight line from the coordinate axis of the images and the pitch, yaw, and role of the aircraft. Unless the acquisition platform is capable of acquiring perfectly nadir imaging on a routine basis, it is necessary to rectify the image or correspondences to enable linear 3D structure estimation.

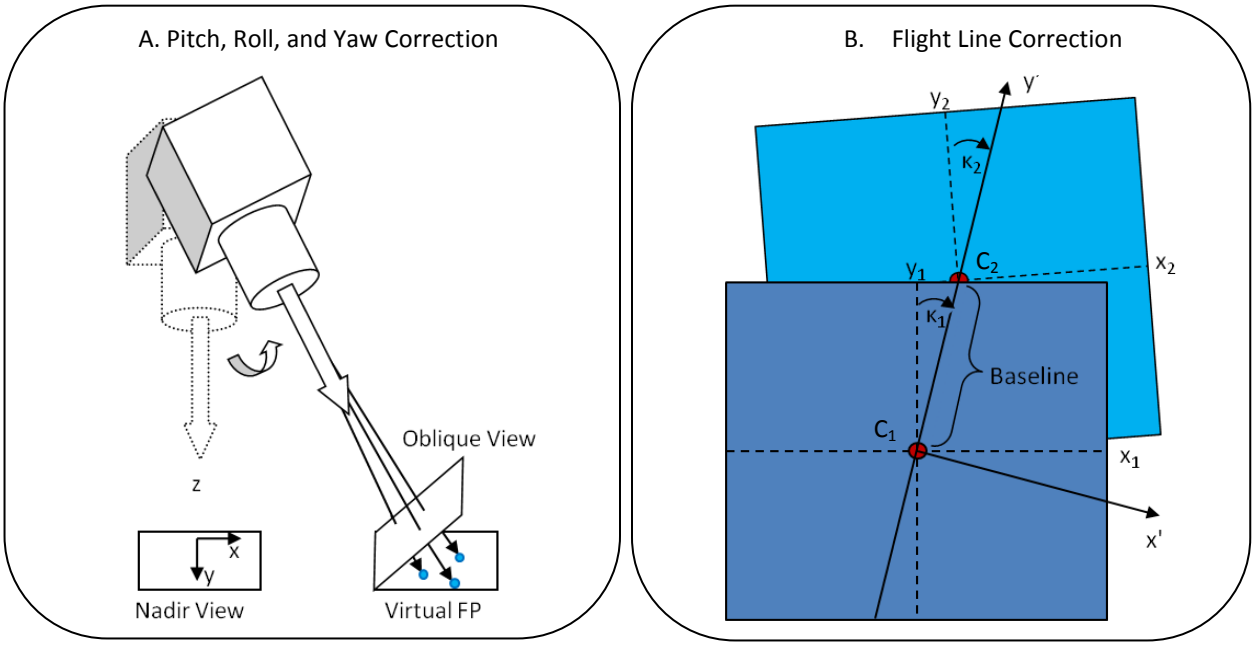


Figure 4-15 Corrections are required to compensate for aircraft pitch, yaw, and roll and flight line orientation as discussed earlier in Section 4.2.1.3. These are done by projecting the matches onto a virtual focal plane and then transforming them to a coordinate system aligning the x-axis to the flight line connecting the two image centers.

The approach the author has taken to accomplish this is to project the image correspondences onto a virtual focal plane that is located at the focal length ***(f)***, but, is situated parallel to the earth's surface as depicted in Figure 4-15. This can be accomplished by using the image projection versions of the collinearity equations below (DeWitt and Wolf 2000), where ***m*** is the rotation matrix, ***(X_L, Y_L, Z_L)*** is the camera location, ***(x_0, y_0)*** is the principal point, ***(x, y)*** is the image location and ***(X, Y, Z)*** is the object location in the WCS.

Collinearity Eq. x-axis image proj.

$$x - x_0 = -f\left[\frac{m_{11}(X - X_L) + m_{12}(Y - Y_L) + m_{13}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)}\right] \quad \textbf{(50)}$$

Collinearity Eq. y-axis image proj.

$$y - y_0 = -f\left[\frac{m_{21}(X - X_L) + m_{22}(Y - Y_L) + m_{23}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)}\right] \quad \textbf{(51)}$$

The flight line corrections can be implemented by generalizing Equations (45) and (46) to accommodate baselines that are offset from the image axes. It is important to note that the height estimate ($\boldsymbol{Z_i}$) is dependent on the ratio of the baseline (***B***) to the pixel distance ($\boldsymbol{p_i}$) of the match points projected onto the virtual focal plane. This ratio can be corrected to one that is aligned with the flight line by performing a coordinate system conversion to the aircraft flight line or by compensating for the relative Baseline distance with respect to the pixel correspondence distances (Equations (52)-(53)). Finally, the corrected image plane distance can be calculated by utilizing Equation (53) with the previous modifications. Here, the offset from the flight line is represented by ***K***.

Baseline Distance Correction

$$B = \sqrt{|cosK|(C_{x2} - C_{x1})^2 + |sinK|\left(C_{y2} - C_{y1}\right)^2} \quad \textbf{(52)}$$

Image Distance Correction

$$p_i = \sqrt{(x_{2i} - x_{1i})^2 + (y_{2i} - y_{1i})^2} \quad \textbf{(53)}$$

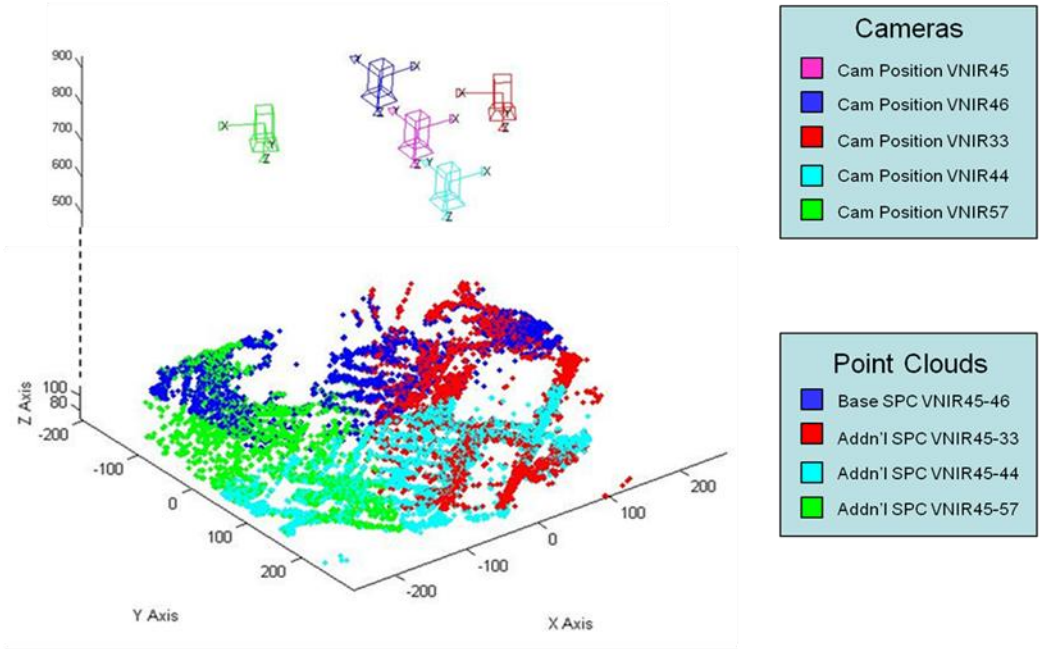


Figure 4-16 The interim estimates of the four individual SPC's can be seen compared to the camera locations.

Interim results can be viewed with their respective camera stations in Figure 4-16, where nearly 20,000 individual point correspondences were automatically recovered from 5 matching images (4 image pairs) to produce a Sparse Point Cloud (SPC) representation of the scene. Note that here the results are still in a relative (meter-based) coordinate system centered on the base camera location.

4.3.2.5 *Non-Linear Optimization of Sparse Structure*

Many of the problems presented in this research cannot be solved by linear methods alone. In these cases, it is necessary to apply non-linear estimation techniques to provide accurate

solutions. Such real-world problems as the resectioning of images to models and the bundle adjustment (BA) of multiple images, to reconstruct 3D structure, both require nonlinear minimization solutions. In fact, for BA, these solutions often depend on calculating the interaction of several thousand variables simultaneously. Due to its stability and speed of convergence, the Levenberg–Marquardt algorithm (LMA) is one of the most popular approaches routinely utilized to solve these challenging problems (Lourakis & Argyros, 2004).

When implementing LMA, the computational challenge is to minimize a given cost function. For applications such as resectioning and BA, this cost function is defined as the sum of the squared error between image points (actual data) and projected 3D model points (predicted values) dictated by the current set of parameter ($\hat{X}$). The minimization function takes advantage of the relationship between the estimated 3D structure ($\bar{\boldsymbol{X}}_i$) and its 2D projection onto the image plane ($\bar{\boldsymbol{x}}_i$) as mathematically formalized below (Hartley & Zisserman, 2004).

Projection Function

$$\bar{x}_i = P\bar{X}_i \tag{54}$$

Projection Matrix

$$P = KR[I \mid -t] \tag{55}$$

The projection matrix (***P***) can then be utilized directly for minimization since it incorporates the cameras internal calibration parameters (***K***), and external orientation (***R)*** and position (***t***). This minimization equation then takes the following form (Equations (56) and (57)), where ***d*** is the Euclidean distance between the image coordinate $\bar{\boldsymbol{x}}$ and the projected 3D point $\bar{\boldsymbol{X}}$.

Projection Minimization Function

$$\sum_{i} d(\bar{x}_i, P\bar{X}_i)^2 \quad \textbf{(56)}$$

Expanded Minimization Function

$$\sum_{i=1}^{n} \left\| \bar{x}_i - \hat{X}_i(K, R, t, \bar{X}_i) \right\|^2 \quad \textbf{(57)}$$

The sparse bundle adjustment (SBA) algorithm of Lourakis and Argyros (Lourakis & Argyros, 2004) is optimized for speed and efficiency. It can easily minimize against several camera variables and the structure of tens of thousands of 3D points simultaneously to produce a sparse image bundle that is mutually self-consistent. However, as with any engineering code, it requires specific formatting for the input variables and special care when preparing the camera's internal and external orientation parameters. The next section addresses this topic in order to ensure that accurate global coordinates can be obtained after utilizing this SBA minimization algorithm.

4.3.2.6 *Relating the Results to World Coordinate System*

Since the results of the SBA process minimize against a relative coordinate system anchored on the base camera position, it can be difficult to determine the absolute locations of the 3D points even though there is good self consistency between the camera locations and the SPC. In order to recover the absolute location of the 3D points, the collinearity equations (Equations (58)-(59)) were utilized to re-project the 3D points back into the base image locations of the initial feature matches as seen in Figure 4-17B.

Collinearity Eq X-component World Coord.

$$X - X_L = (Z - Z_L)\left[\frac{m_{11}(x - x_0) + m_{21}(y - y_0) + m_{31}(-f)}{m_{13}(x - x_0) + m_{23}(y - y_0) + m_{33}(-f)}\right] \quad \textbf{(58)}$$

Collinearity Eq Y-component World Coord.

$$Y - Y_L = (Z - Z_L)\left[\frac{m_{12}(x - x_0) + m_{22}(y - y_0) + m_{32}(-f)}{m_{13}(x - x_0) + m_{23}(y - y_0) + m_{33}(-f)}\right] \quad \textbf{(59)}$$

In this case, only the minimized depth parameter (***Z$_i$***) retained its absolute coordinate value and so could be utilized with the camera locations ***(X$_L$, Y$_L$, Z$_L$)*** to determine the world coordinate latitude (***Y$_i$***) and longitude (***X$_i$***) values. The final results are display below in Figure 4-17 showing the UTM SPC (A), a facetized height map (B), in Google Earth as individual 3D points (C) and re-projected back into the base image to show how a UV Texture Map can be derived (D).

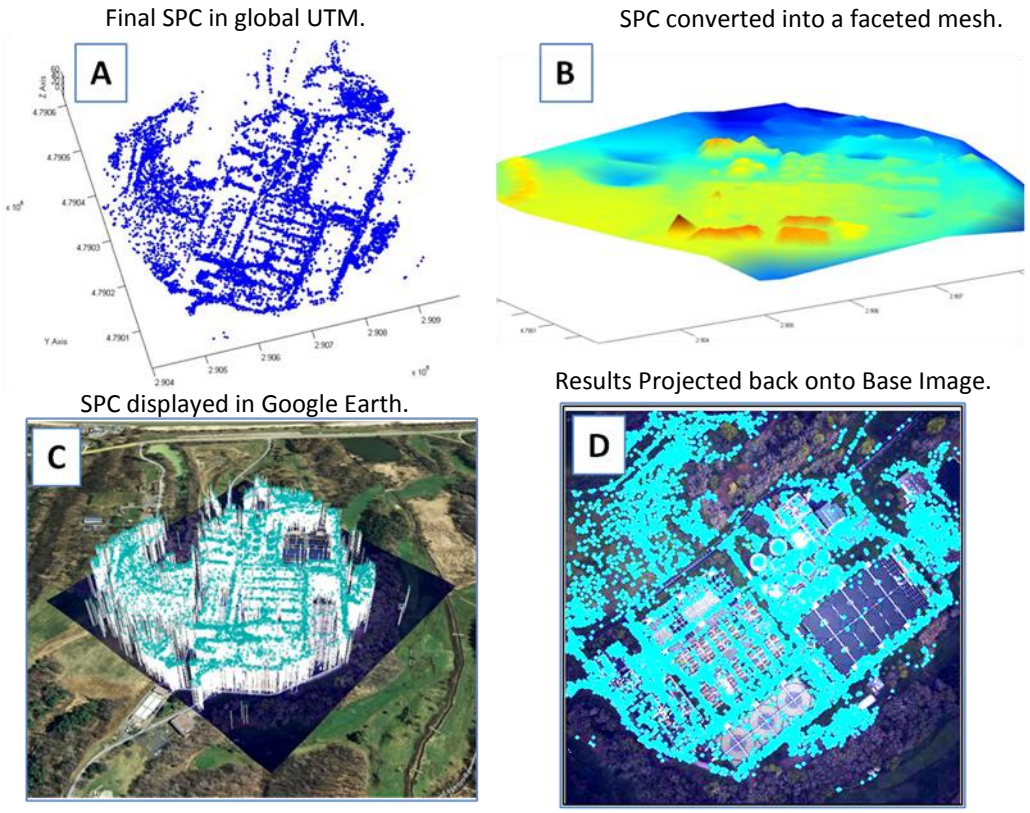


Figure 4-17 Example results of the Sparse Bundle Adjustment process on the Sparse Point Cloud. Here the absolute global coordinates (A) can be compared to the facetized surface (B), visualized in Google Earth (C), or re-projected back into any of the images contained within the bundle (D).

Below, a comparison of the final image derived SC mesh can be compared to a standard 30 [m] Digital Elevation Map (DEM) and to a hi-fidelity 1 [m] LIDAR Terrain Elevation Map (TEM).

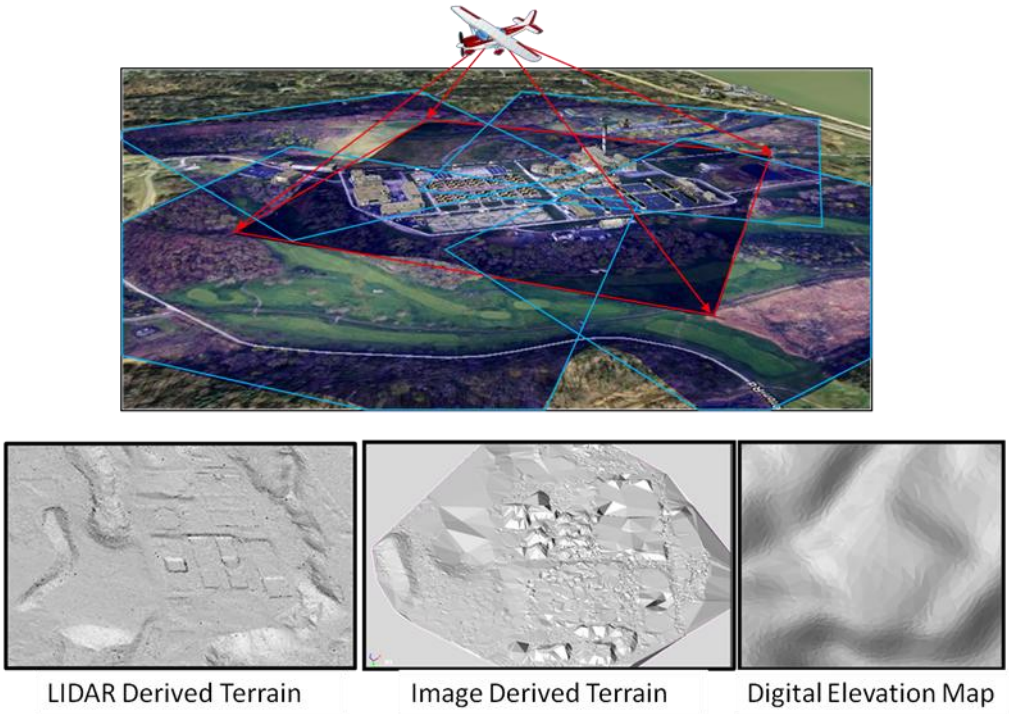


Figure 4-18 The image derived SPC mesh fidelity can be directly compared to both hi-fidelity ~1 [m] LIDAR terrain and a lo-fidelity ~30 [m] Digital Elevation Map.

Relating the SBA-SPC to the WCS comes with the understanding that the final product will only be as accurately positioned as the accuracy of the IMU-GPS information available from the flight recorder. Since the SPC is minimized against a the position of all the cameras relative to a base camera position, the results are only as good as the position and orientation accuracy of your base camera. For a closer look at how these results can be compared and registered to a LIDAR dataset, see Section 5.1.3.

4.3.3 Recovering Dense Structure from Images

For completeness and to show the "end game" of recovering detailed scene models solely from imagery, this section gives a cursory introduction of some of the related work accomplished with PhD Student Dave Nilosek. Some of his initial work, into the recovery of Dense Point Clouds from multiview imagery, is represented here.

The key to recovering a Dense Point Cloud (DPC) from matching images lies in the ability to relate the images on a pixel-to-pixel level (Nilosek, et al. 2009). This is the transition point between the macro and micro scene reconstructions. Here the micro process requires certain information derived from the macro process to optimally utilize the derived mathematical relationships between the images and the SPC. At this point in the scene reconstruction, each image is already related to a base image of the scene through a fundamental matrix and the SPC is related to each image using a projection matrix. The macro process has also derived the regions of overlap for each image with respect to the base image. Each fundamental matrix, projection matrix and region of overlap is passed off to the micro process with the SPC. Ideally the micro process would relate every pixel in every overlapping image to the base image; however, due to computing power restrictions, examples in this paper focus on specific targets inside the regions of overlap.

4.3.3.1 *Dense Correspondence - Relating Images at the Pixel Level*

The utility of the fundamental matrix for outlier match removal has already been shown, now this matrix will be used to help derive a dense set of matches between overlapping regions. Using this matrix and Equation (41) for every pixel in the base image, an epipolar line that

contains the corresponding point can be found in each overlapping image. Figure 4-19 shows how epipolar lines are found in different overlapping regions from a single point in one image for three different images.

Figure 4-19 Left: Image with single point chosen. Middle/Right: Corresponding epipolar lines in other images.

This property of the fundamental matrix reduces the correspondence search to a one-dimensional search along epipolar lines. The images are rectified so that the epipolar lines run along the horizontal and then a normalized cross correlation is computed based on a small area selected around the target pixel in the base image. The maximum response from the normalized cross correlation is chosen as the match. This is done for every pixel over the entire area which results in a very dense correspondence between the multiple views.

The estimate of the dense structure follows the same pipeline as estimating the sparse structure. First basic photogrammetry is used to extract an initial estimate of the structure. Then the camera parameters, initial estimate of the structure and correspondences are used in minimizing the reprojection error between all the images using the SBA method. The collinearity equations can also be used to place the dense structure in the world coordinate system. Additionally, the dense structure can be texture mapped with an image of the target as shown in Figure 4-20.

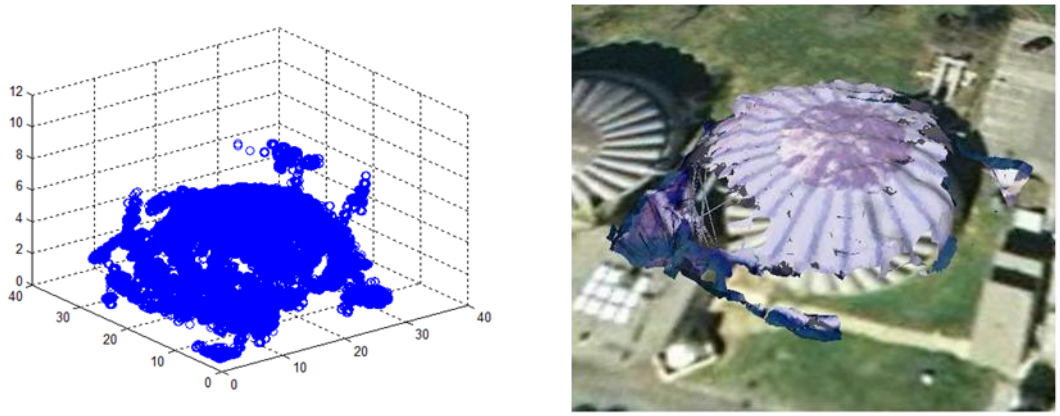


Figure 4-20 Left: Initial estimate of the structure of the dense point cloud from three images. Right: Result after SBA, world coordinate mapping and projective image texturing.

The initial estimate of the structure and the final product is shown in Figure 4-20 after all the steps are completed. Once the dense structure of a specific target has been acquired, it is combined with the sparse structure. Figure 4-21 shows the dense structure incorporated into the sparse structure and overlaid on a map. Also on this map are image-derived, but, manually generated CAD models of similar structures in the scene (Pictometry, 2008). The automatically generated dense structure can now be directly compared to the structure of the CAD model for verification. One very clear issue still remains when working with only nadir imagery and that is the difficulty in reconstructing the sides of objects. Although oblique imagery can be used to view the vertical detail of the scene, the severe projective transforms that relate these images can provides additional correspondence challenges which are discussed below.

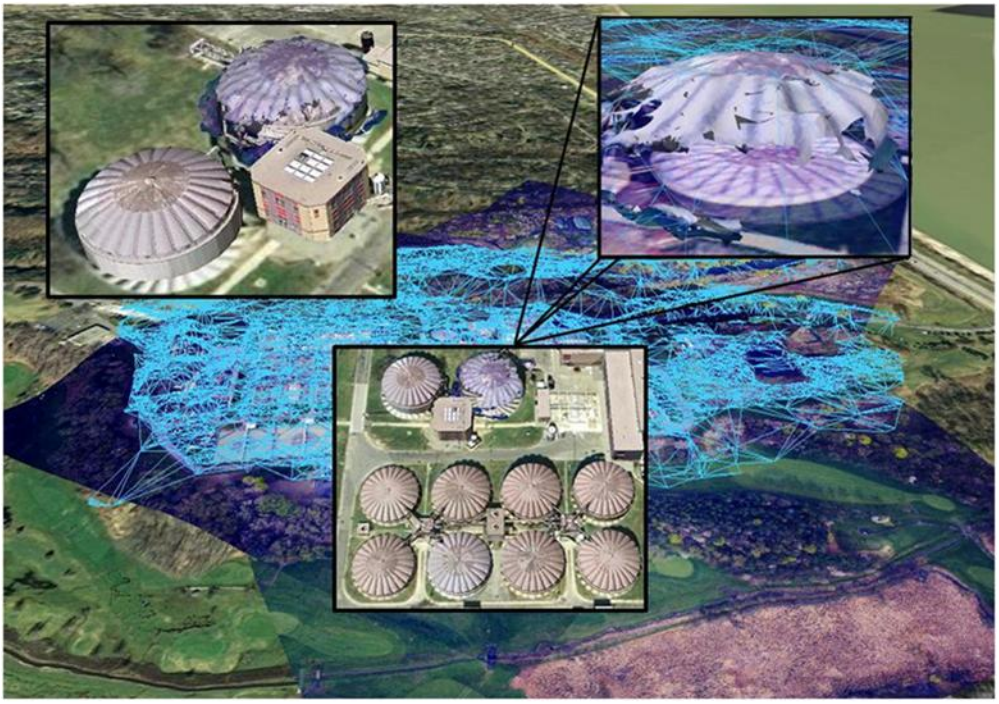


Figure 4-21 Resulting 3D structure recovered from three overlapping images using Dense Point Correspondences (The model provided by Pictometry is embedded within Google Earth).

4.3.3.2 *Matching Oblique Images using ASIFT – Maximizing Angular Diversity*

Recently an algorithm has been developed that attempts to describe features as projectively invariant. This algorithm is called Affine Scale Invariant Feature Transform (Morel & Yu, 2009). This algorithm builds on the original SIFT algorithm by taking the initial images and simulating rotations along both the x and y axis. It essentially performs many SIFT operations over these simulated images in order to find the best matching rotation between the images in order to remove it. Once the initial matching is found using ASIFT, the same RANSAC process, using the fundamental matrix as the fitting model, can be used to eliminate the outliers. Figure 11 shows an example of matching points using ASIFT and then RANSAC.

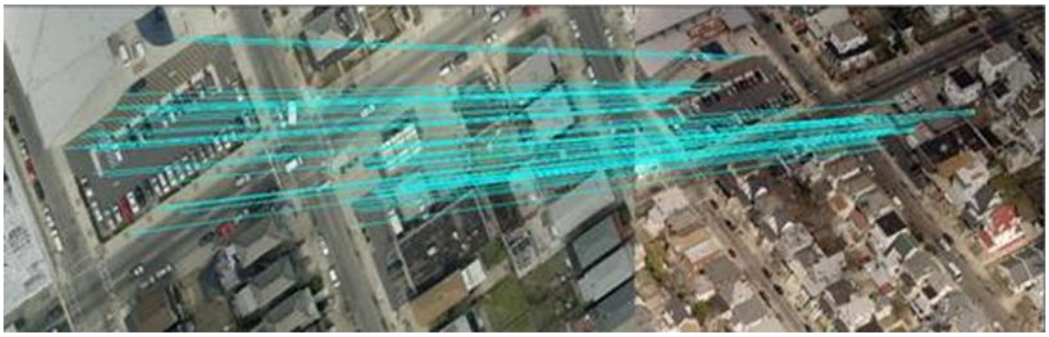


Figure 4-22 Matching between a nadir and oblique images using ASIFT and then RANSAC with the Fundamental Matrix as the fitting model (Images courtesy Pictometry Int. (Pictometry 2010)).

The next step is to utilize the SPC, resulting fundamental matrices and regions of overlap to extract a DPC of a target area within the scene. Since a projective transformation can greatly impair the normalized cross-correlation method of point matching, other approaches may be required for dealing with images that capture significant angular diversity of a target.

4.3.3.3 *Growing a Depth Map from Sparse Correspondences*

Since an accurate sparse representation of the structure of the scene has already been derived, this structure can be utilized as a good starting point to 'grow' a dense match between images. (Goesele, Snavely, Curless, Hoppe, & Seitz, 2007). A dense matching is generated around each sparse match using an optimization method that minimizes the normalized pixel intensity difference between each overlapping image with respect to the base image. Here each projected SPC location is utilized as an initial seed and the matched image locations are slowly grown from the pixels surrounding these points. In this way a dense correspondence mapping can be obtained between images by constraining the epipolar line search space.

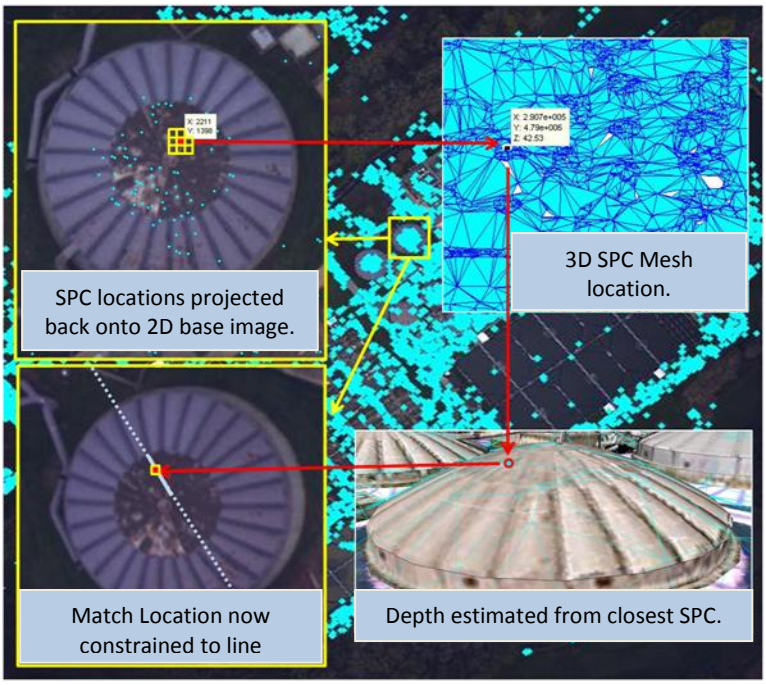


Figure 4-23 Growing 3D depth maps based on the initial SPC results and epipolar relationships. In the upper left inset, the 3D SPC is projected back onto the base image. For these locations the depth information is already known (upper right) and can be used to constrain the matching locations in the other images (lower left) to follow a general surface function.

4.3.4 AeroSynth Summary

Due to the fast growth in the computer vision arena, regarding SfM techniques (Chapter 14), it is fruitful for the photogrammetry community to keep abreast and apply these techniques to the area of remote sensing. The AeroSynth technique for recovering 3D structure from images is a blend of both photogrammetric and computer vision approaches. It utilizes the automatic feature isolation/matching, epipolar relationships and SBA of the computer vision community and combines it with the linear 3D point estimation and collinearity relationships of photogrammetry. As a result, the image bundle, SPC, and DPC that is produced can be related to the WCS and directly injected into GIS applications for automatic analysis and comparison to existing archival data.

4.4 Sparse Bundle Adjustment (SBA)

Once the initial estimates for a camera's EOPs/IOPs and the 3D structure of the image correspondences have been produced, a Bundle Adjustment (BA) process is commonly used to bind these results into a self-consistent solution. Since this can be geometrically visualized as a bundle of rays which piercing the image plane on their path from the 3D object through the camera lens, it has been commonly referred to as a "Bundle Adjustment". The "Sparse Bundle Adjustment" name comes from the sparse nature of the matrices involved in its solution and not from the Sparse Point Clouds that can be generated through the BA process.

> "As an indication of its efficiency, it is noted here that one of the test problems to which SBA has been applied involved 54 cameras and 5207 3D points that gave rise to 24609 image projections. The corresponding minimization problem depended on 15999 variables … without a sparse implementation of BA, a problem of this size would simply be intractable." -(Lourakis and Argyros 2009)

The task presented in the last section, regarding minimization of the unknown parameters, can become very challenging. This is due to the fact that the $11 \times m + 3 \times n$ total parameters must now be factored in a Jacobian matrix that has $(11 \times m + 3 \times n) \times (11 \times m + 3 \times n)$ variables, which becomes impractical without implementing sparse matrix techniques for a solution. The figures below (Figure 4-24, Figure 4-25, & Figure 4-26), should give the reader an appreciation of the sparse structure of the solutions space, which is due to the general lack of interdependence of the variables which are being solved. Please reference Chapter 13, for a more detailed review of how the SBA is solved using the Levenberg-Marquardt Algorithm (LMA).

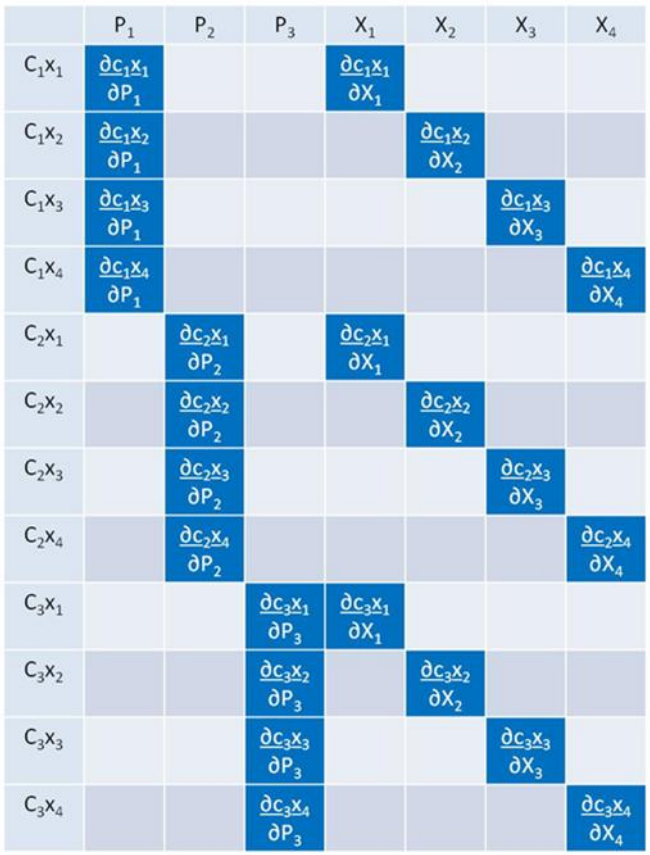


Figure 4-24 The structure and composition of a Bundle Adjustment Jacobian matrix.

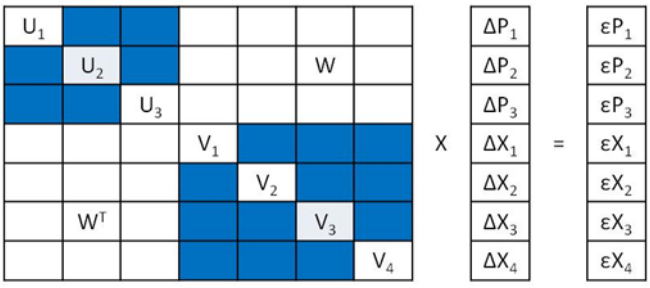


Figure 4-25 The structure and composition of the normal equations (~Hessian matrix).

For these problems the following equations hold ((Hartley and Zisserman 2004) and are graphically represented in Figure 4-25.

$$U_j = \sum_i \left(\frac{\partial x_{ij}}{\partial P_j}\right)^T \Sigma_{x_{ij}}^{-1} \left(\frac{\partial x_{ij}}{\partial P_j}\right) \quad \textbf{(60)}$$

$$V_i = \sum_j \left(\frac{\partial x_{ij}}{\partial X_i}\right)^T \Sigma_{x_{ij}}^{-1} \left(\frac{\partial x_{ij}}{\partial X_i}\right) \quad \textbf{(61)}$$

$$W_{ij} = \left(\frac{\partial x_{ij}}{\partial P_j}\right)^T \Sigma_{x_{ij}}^{-1} \left(\frac{\partial x_{ij}}{\partial X_i}\right) \quad \textbf{(62)}$$

$$\varepsilon_{P_j} = \sum_i \left(\frac{\partial x_{ij}}{\partial P_j}\right)^T \Sigma_{x_{ij}}^{-1} \varepsilon_{ij} \quad \textbf{(63)}$$

$$\varepsilon_{X_i} = \sum_j \left(\frac{\partial x_{ij}}{\partial X_i}\right)^T \Sigma_{x_{ij}}^{-1} \varepsilon_{ij} \quad \textbf{(64)}$$

$$\Delta P_j = P_j - P_{j-1} \quad \textbf{(65)}$$

$$\Delta X_i = X_i - X_{i-1} \quad \textbf{(66)}$$

The sparse form of the Hessian becomes very apparent in Figure 4-26, for large numbers of 3D points and camera parameters.

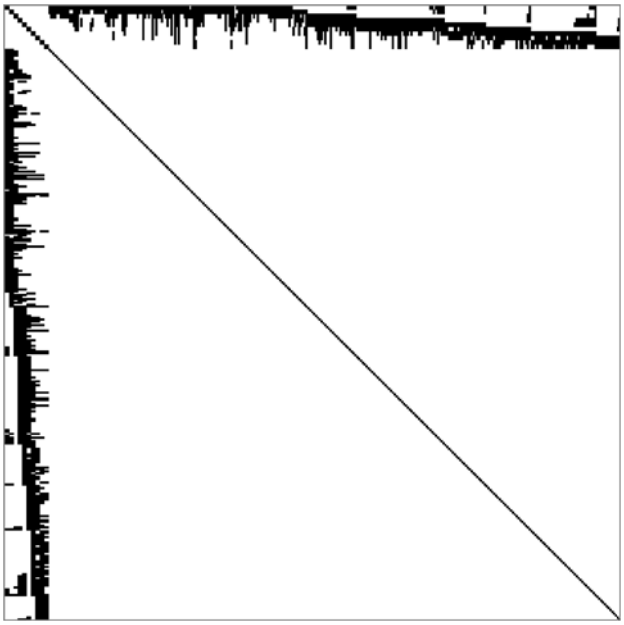


Figure 4-26 A sparse matrix obtained when solving a modestly sized bundle adjustment problem. This sparsity pattern is of a 992x992 normal equation (i.e. approx. Hessian) matrix, where black regions are nonzero blocks. (Lourakis and Argyros 2009)

It should be noted that minimizing the linear estimate of our camera projection models (P_i) by utilizing the matching 2D image point correspondences is exactly the same process involved in the camera resectioning task of Section 3.2. This is convenient, because some of the mathematical infrastructure necessary to accomplish our task for BA has already been developed. Only the 3D estimate for the point cloud (Section 4.2.1.4) and inclusion of the minimization parameters against those points are required to implement the BA process.

As mentioned earlier in this chapter, the SBA code of Lourakis and Argyros (Lourakis & Argyros 2010) was utilized to accomplish the minimization due to its speedy C++ implementation and proven performance. This is a good alternative for implementation of potentially large datasets that could be derived from numerous flights over the same target area. Additionally, a MATLAB interface is available which was incorporated and modified to be used with the WASP sensor EOP and IOP flight information. The main drawback is that like any engineering C++ code, it represents a "black box" solution that can only be partially modified for research. Additionally, the documentation about how the orientation angles were implemented was noticeably lacking, although this appears to have been addressed in the most recent version now available. In hindsight, a purely MATLAB implementation would have been a more flexible tool for academic use.

In the next chapter, we will again add dimensionality to the data relationship challenge by relating 3D rigid bodies. Many of the previous concepts will be utilized and expanded to address the additional requirements of deriving a purely 3D solution for these problems.

5 Relating Rigid 3D Bodies

To register 3D data, such as a Sparse Point Cloud (SPC) to a Dense Point Cloud (DPC) or Faceted Model (FM) will require slightly different techniques compared to the previous chapters. Of primary difference is the process by which we can extract common invariant features and relate them via 3-D matching techniques. Additionally, the final transform, that will be utilized to relate the datasets, is no longer constrained to a 2D projection of the 3D model. It is now a fully 3D transformation that may be constrained to rigid-body solutions.

The potential payoff, for developing sound SPC to DPC/FM registration techniques, is that we will be able to utilize the global coordinate system of our LIDAR or Model data, to orient our locally related bundle of images, cameras, and SPC. This is potentially the area of highest customer interest, since there is currently much growth in both the online FM generation (Google Sketchup 2009) and local sparse structure development using tools like PhotoSynth (Microsoft Corporation 2010) and no current way to easily relate the two environments. In fact, the originators of the PhotoSynth process (Snavely, Seitz and Szeliski, Photo tourism: Exploring photo collections in 3D 2006), utilized primarily manual processes to relate their Sparse Structure Bundles (SSB) to terrain maps.

As with the 2D-to-2D and 2D-to-3D registration, the process for 3D-to-3D registration is to:

a. *Extract similar invariant features*

b. *Match these features*

c. *Utilize these Correspondences to create a Mathematical Relationship*

If this can be done robustly and accurately, the 3D data can be related. Figure 5-1 details the steps for 3D registration in pictorial form.

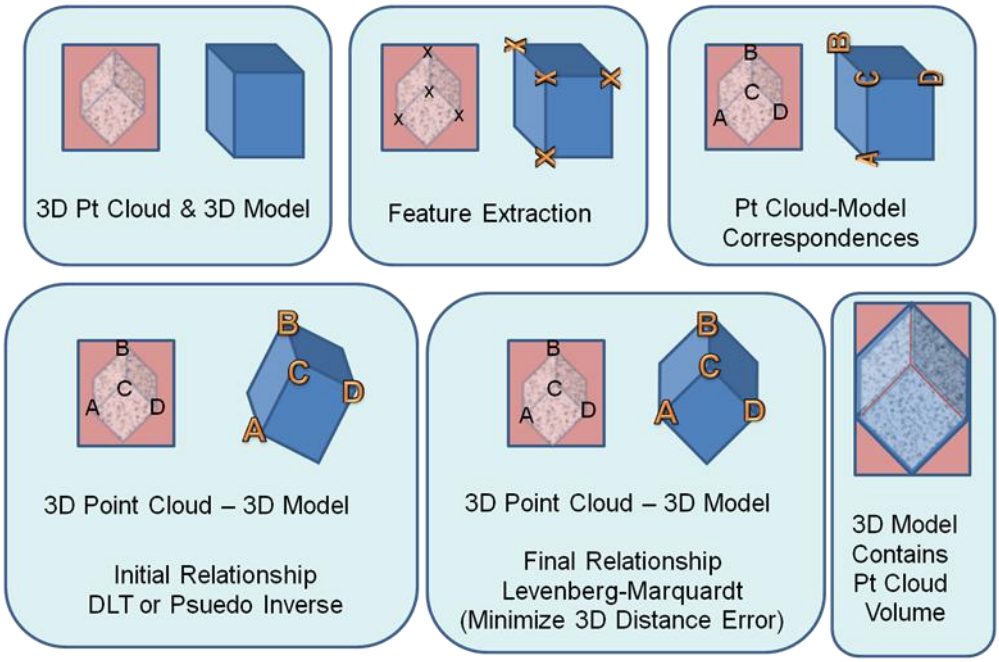


Figure 5-1 The basic process for relating 3D models and structure using a 3d Conformal transform. As in the previous sections, the key here is to relate similar features within the two datasets in order develop a mathematical relationship. The only added complexity is in the additional dimensionality and possible feature disparity of the datasets.

5.1 Sparse to Dense Point Clouds

This section addresses the unique challenge of relating the SPCs developed by an SBA algorithm and the DPCs that are common to LIDAR data. This challenge is unique, because there is little current research into the automated matching of SPCs features which is known to the author beyond the Iterative Closest Point algorithm used with LIDAR DPCs. Additionally, the problem may be ill-posed if there are no common elements from the datasets. In this case, estimation may be limited to the statistical analysis of point distributions w.r.t regional densities and their inter-relationships.

The SPC will often be the result of an SBA process, similar to the one developed in Chapter 4 and will range from a few hundred points to tens of thousands, depending on the number of images that were related. The DPC will be the result of a LIDAR data collection or the dense correspondences resulting from a model reconstruction (Pollefeys, et al. 2004) and will normally range from hundreds of thousands of points to millions. Examples of an SBA-SPC and LIDAR-DPC of the Midland, MI power plant can be viewed below in Figure 5-2.

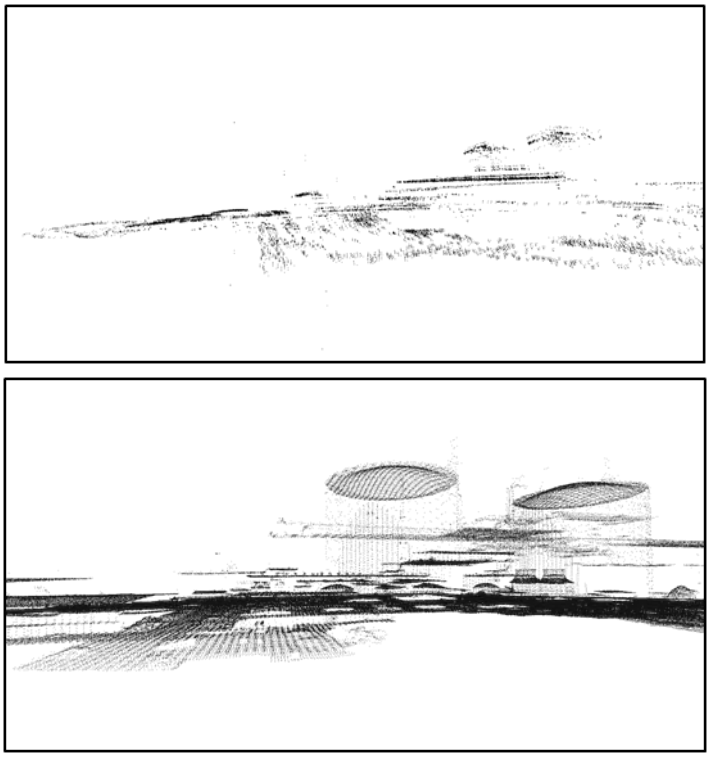


Figure 5-2 The Midland Site SPC (top) resulting from BA of tens of thousands of 3D points compared to the millions of 3D points embedded within a LIDAR DPC (Bottom).

In either case, the distinction between SPCs and DPCs is rather vague and only used here to distinguish between the amount of data available for registration. The key is in the determination of common structure elements, if any exist, between the two datasets.

5.1.2 Approach

When trying to relate SPCs to other data, the features of interest may have already been identified; meaning that we may want to initially utilize every point in the sparse structure for potential correspondence. Additionally, we hope that the DPC has a subset of points that are common to the SPC that can be culled for matching. So, if 3D point correspondence matching is feasible, feature extraction may be relegated to an analysis of which features in a DPC (such as LIDAR data) will correlate to those extracted in the SPC creation process. The following are few of the techniques implemented by the author.

5.1.2.1 *Using Global Coordinates*

If the both the SPC and DPC datasets are described in a real-world coordinate system, then a straightforward implementation of the Iterative Closest Point (ICP) algorithm(Z. Zhang 1992) can normally provide adequate correspondences for a mathematical description. This entails minimizing the distance between every SPC point and the nearest n DPC points; where robust values for n can be gauged based on the volumetric point density (Figure 5-3).

5.1.2.2 *Using Relative Coordinates and User Assistance*

For every 3D pt in an SPC, a corresponding region within a DPC can be identified with a sphere encircling several DPC points. Thus when relating the two datasets, it is possible to isolate the "best" correspondence within a DPC by minimizing the distance among the closest regional points. This approach is straightforward to implement, since a user can easily identify regions that an SPC point may relate to in a DPC and wouldn't have to worry about precise correspondence determination (Figure 5-3).

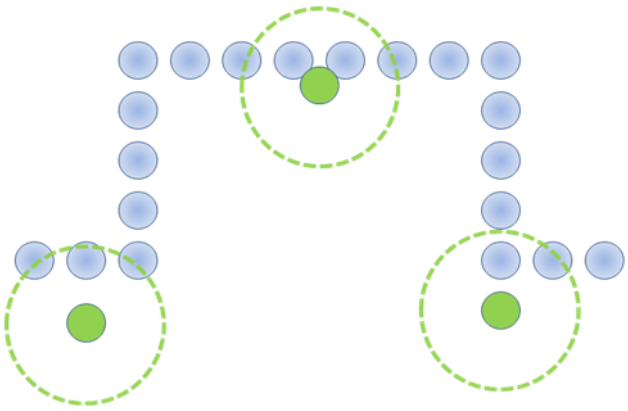


Figure 5-3 Relating the SPC pts to DPC points via an iterative nearest neighbor approach.

A close approximation to this technique is one in which both the SPC and the DPC have been facetized and the user selects similar locations on the models. This may, or may not, fall in close proximity to an existing model vertex location. However, the author has found that this technique is both easy to accomplish and provides good results for relating these very different types of models. For this reason it was utilized in the case study below.

5.1.3 Case Study

The case study described below provides new research into an area that is sure to get much attention in the future. This is in the challenging area of relating Sparse Point Cloud Models to other imagery derived products and especially LIDAR derived Dense Point Cloud models.

SBA-SPC to LIDAR-DPC: The data set used here is the SBA-SPC that was developed in Chapter 4 and a LIDAR-DPC (Kucera International Inc. 2010) of the VanLare Water Processing Plant, Rochester, NY that was created using the author's MATLAB facetization algorithm.

In this case study, the analysis was performed against the local MegaScene Tile-4 Location and the initial results of a 3D RMSDE calculation resulted in an average error of ~11 [m].

Table 1 – This table shows the initial error of the faceted SPC results when compared to their matching features in a faceted LIDAR model of the same location, these matches can be visualized in Figure 5-4.

Control	LIDAR Vat CPs Local [m]:			SPC Vat CPs Local [m]:			Error Calculations [m]			
Points	x	y	z	x	y	z	x1-x2	y1-y2	z1-z2	RMSDE
1	853.32	435.81	103.09	837.50	431.39	111.94	15.83	4.42	-8.85	10.78
2	873.02	463.74	103.13	857.76	459.92	110.24	15.26	3.82	-7.10	9.96
3	905.22	425.73	99.59	888.79	421.18	108.50	16.43	4.55	-8.91	11.10
4	920.54	447.88	99.37	902.77	444.07	106.50	17.77	3.81	-7.13	11.27
5	936.48	470.39	99.24	919.02	465.74	105.00	17.46	4.66	-5.76	10.95
6	912.42	387.36	99.59	894.95	384.17	108.82	17.47	3.19	-9.23	11.55
7	927.74	410.37	99.38	912.19	406.52	107.46	15.55	3.84	-8.08	10.36
8	943.03	432.00	99.45	925.60	428.10	106.91	17.43	3.89	-7.45	11.17
9	958.38	454.78	99.31	943.00	450.13	105.64	15.38	4.66	-6.32	9.97
						Ave RMSDE [m]:	16.51	4.09	-7.65	10.79

It is important to remember that this is the absolute error, before a simple 3D translation is implemented (using the X, Y, & Z translation error in Table 1) to obtain the final positions of the faceted SPC within the WCS. Once this has been accomplished the error analysis is computed against these new locations as seen below in Table 2.

Table 2 – This table shows how a simple 3D Translation derived from the average error on the 3 axes can be utilized to correct for any residual error in the SPC WCS location.

	LIDAR Vat CPs [m]:			SPC Vat CPs [m]:			Translated Model CPs [m]:			
Control Pt	X	Y	Z	x	y	z	X'	Y'	Z'	RMSDE [m]
1	853.32	435.81	103.09	837.50	431.39	111.94	854.00	435.48	104.30	0.82
2	873.02	463.74	103.13	857.76	459.92	110.24	874.27	464.01	102.59	0.80
3	905.22	425.73	99.59	888.79	421.18	108.50	905.30	425.28	100.85	0.77
4	920.54	447.88	99.37	902.77	444.07	106.50	919.27	448.16	98.85	0.81
5	936.48	470.39	99.24	919.02	465.74	105.00	935.53	469.83	97.35	1.26
6	912.42	387.36	99.59	894.95	384.17	108.82	911.46	388.26	101.17	1.19
7	927.74	410.37	99.38	912.19	406.52	107.46	928.70	410.62	99.81	0.62
8	943.03	432.00	99.45	925.60	428.10	106.91	942.11	432.20	99.26	0.56
9	958.38	454.78	99.31	943.00	450.13	105.64	959.51	454.22	97.99	1.06
					Transformation: 3D Translation				Total Ave RMSDE:	0.88

Once similar points were associated visually, a 3D translation was implemented from the resulting transform that moved the SPC model to its final position.

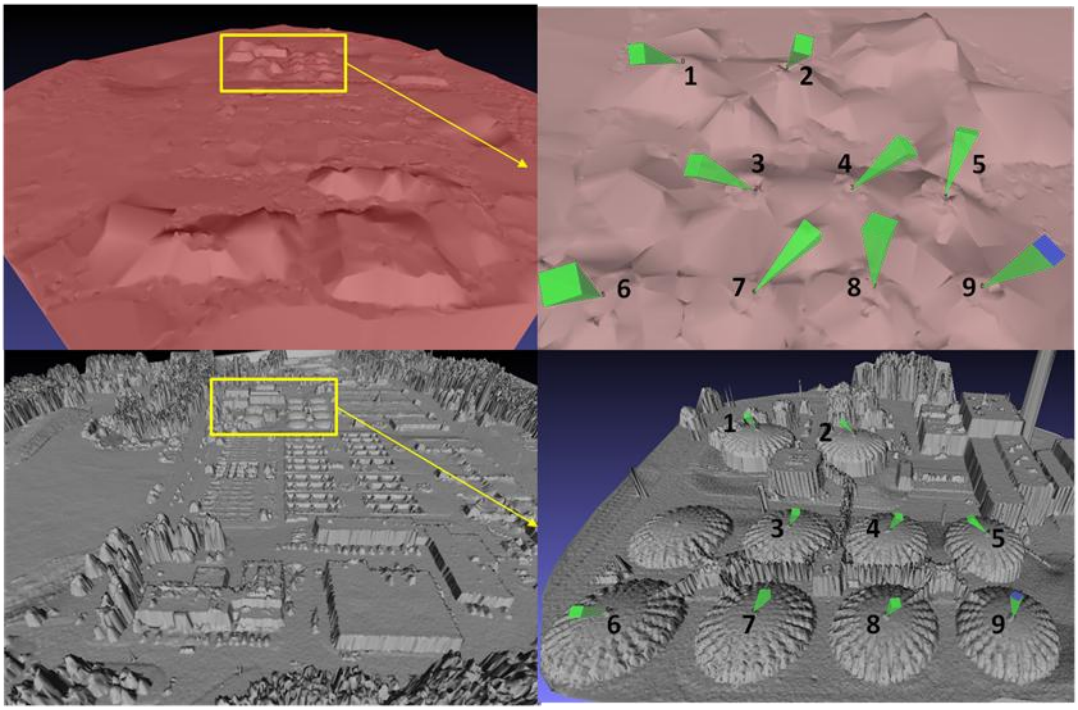


Figure 5-4 The image derived SPC mesh above is compared to a LIDAR derived DPC mesh below for comparison in Meshlab. The absolute coordinates of the image derived results are only as accurate as the projected location of the base image, so a final translation, acquired from the matched locations (right), may be necessary.

This new location corresponds nicely to the LIDAR dataset as seen in Figure 5-5 below. Here the linear 3D Translation ($T_x = 16.51,\ T_y = 4.09,\ T_z = -7.65$) derived from the averaged control point error from Table 1 was utilized in conjunction with a nonlinear refinement using an integrated ICP algorithm within the Meshlab program (Pisa 2010).

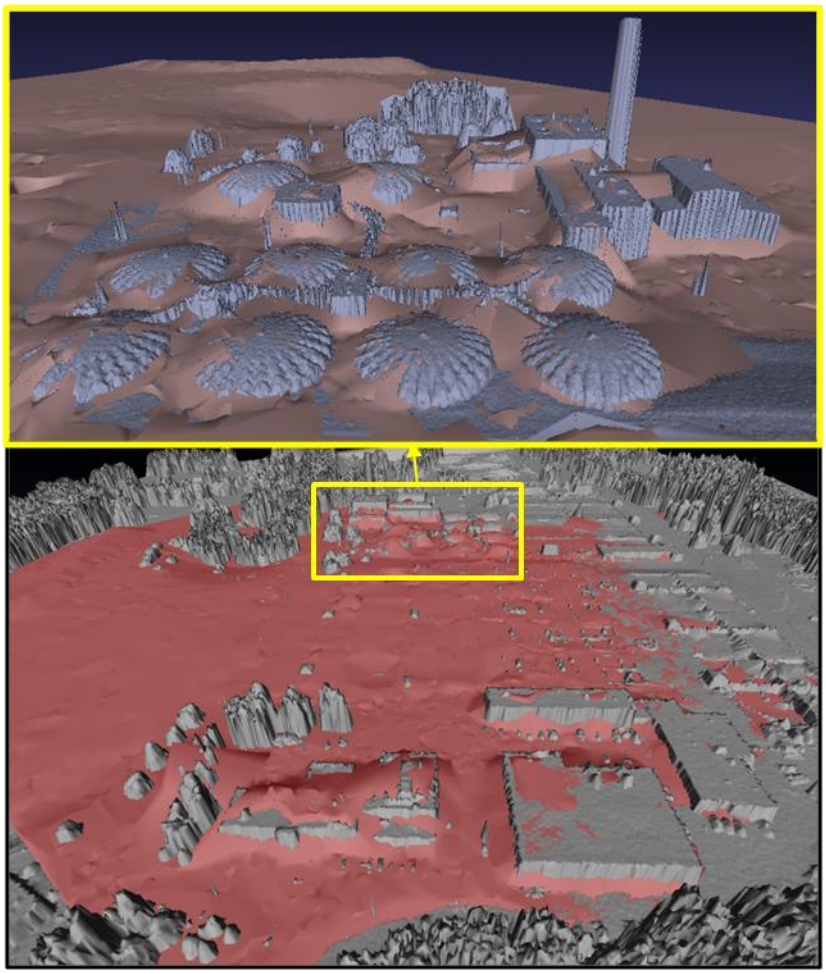


Figure 5-5 The results of the linear 3D Translation and Meshlab (Pisa 2010) implemented ICP nonlinear refinement can be visualized above. Note the general agreement between LIDAR and SPC surfaces as they fight for visibility across the scene.

5.2 Point Cloud to Faceted Model (FM)

The problem addressed in this section is the association of FMs to point clouds. Although the conformal 3D transformation process used to relate the rigid body data will remain the same, the challenge once again is to find adequate correspondences between the often dissimilar datasets.

5.2.1 Approach

Although the correspondence problem is challenging, it is not insurmountable. In fact, the approach implemented by the author here is quite similar to the approach utilized in the last section. The first task is to facetize the Point Cloud, whether it is Sparse or Dense, and then to select similar features correspondences. These features may occur on either related vertices or within an individual facet plane.

5.2.2 Case Study

This case study illustrates the process utilized to take the hi-fidelity faceted model (Pictometry 2010) of the VanLare Water Processing plant, created via manual imagery derived techniques, and relate it to the WCS through a 3D conformal registration with a LIDAR model. All that is required was to accomplish a robust facetization of the LIDAR Dense Point Cloud (Figure 5-6) by using the author's robust facetization code developed in MATLAB.

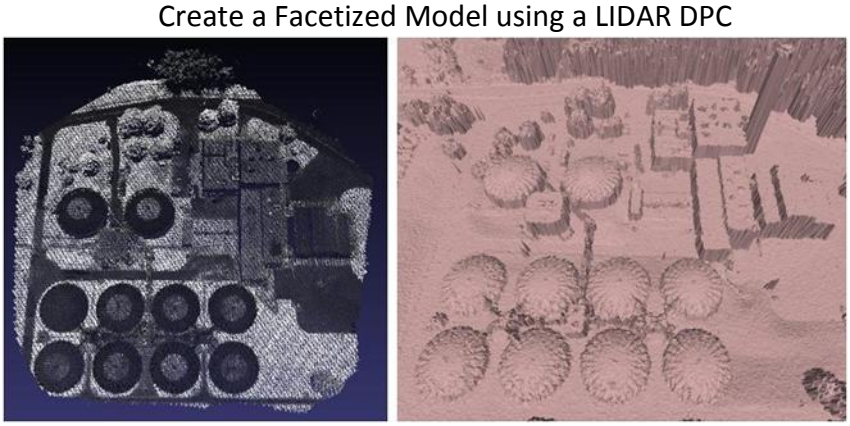


Figure 5-6 This illustrations shows the initial LIDAR DPC with grayscale intensity attributed points on the left. This can be utilized to produce a clean facetized model utilizing the author's MATLAB code as shown in the graphic on the right.

Once this is accomplished, the user must select no less than three nonplanar correspondences (as shown in Figure 5-7) to enable a3D Conformal Rigid body solution to be developed.

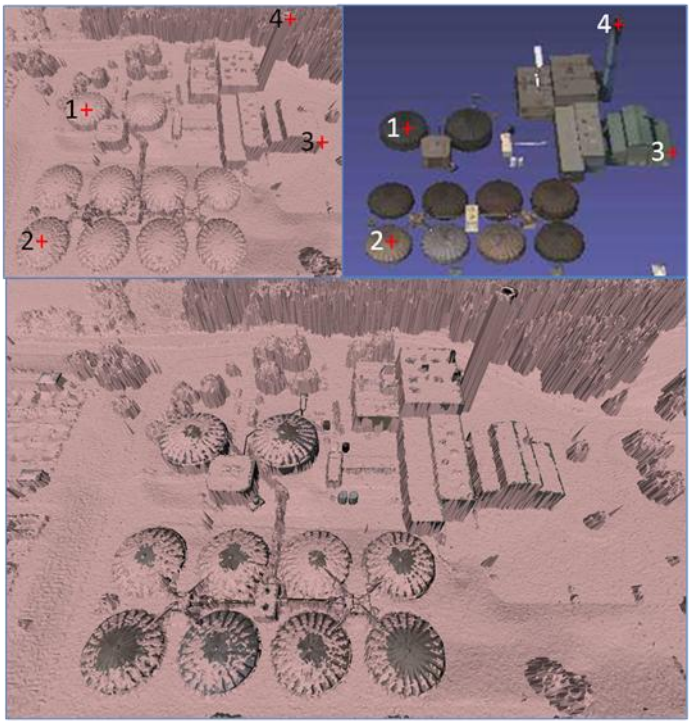


Figure 5-7 This graphic portrays a manual feature correspondence generation that can be used to relate a Faceted Model to a LIDAR DPC that has been facetized. Once accomplished, the initial relationship is improved through nonlinear ICP analysis.

This transform is then applied to the hi-fidelity facetized model to properly position it within the WCS. The following table shows how a simple selection of 4 points from both models can provide a sub-meter accuracy registration of the local model to the LIDAR Data. It should be noted here that the coordinates were converted to the local offset of MegaScene Tile-4.

Table 3 – This table shows the Control Points (CPs) from the LIDAR and hi-fidelity FM which were used to develop a 3D transform to relate the model to the WCS. The New Model CP accuracy is gauged using the 3D RMSDE on right.

	LIDAR CPs [m]:			Model CPs[m]:			New Model CPs [m]:			
Control Pt Location	**X**	**Y**	**Z**	**x**	**y**	**z**	**X'**	**Y'**	**Z'**	**RMSDE [m]**
1) SW Vat	853.6	435.8	103.1	-189.4	1537.5	93.6	853.59	435.81	102.50	0.34
2) SE Vat	911.8	387.7	99.6	-129.7	1583.8	90.6	911.79	387.60	99.21	0.23
3) NE Barn Corner	890.2	540.5	139.5	-155.8	1431.3	130.8	890.40	540.69	139.76	0.22
4) Smokestack	950.1	536.1	97.0	-96.2	1434.6	87.5	949.91	535.94	97.68	0.42
Total Ave RMSDE	Transformation: Conformal 3D Rigid Body									0.30

The transformed model location can then be placed onto terrain that was also developed from the “bare-earth” LIDAR returns (Figure 5-8). In this way the LIDAR Surface Elevation Map (SEM) allowed for a direct relationship with the Faceted Model and then was utilized to create the terrain for that model.

The Transformed Faceted Model placed on LIDAR a Derived Terrain.

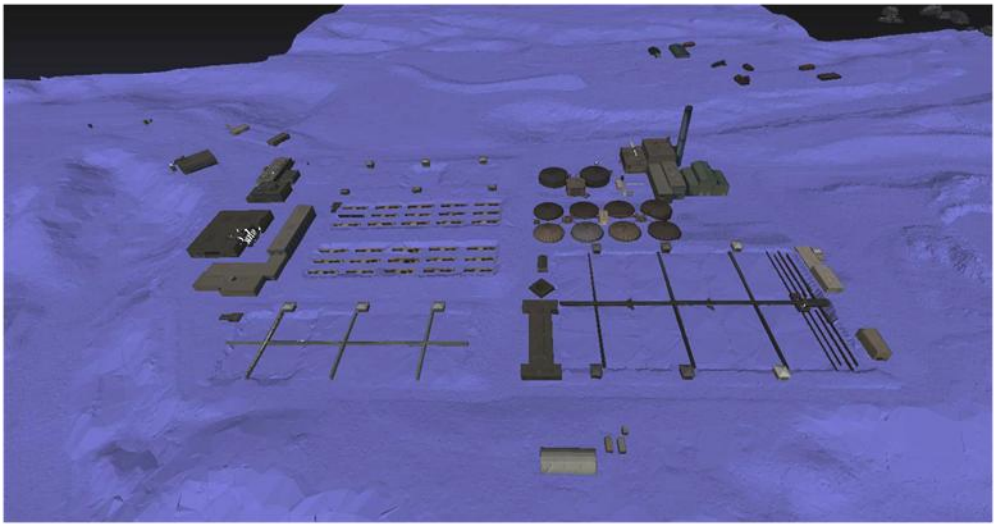


Figure 5-8 The graphic above shows how the Conformally transformed site model can then be placed on the same LIDAR dataset that was now used to create a bare-earth terrain model.

Although the error analysis was performed from the selected point in Figure 5-7 using the authors 3D Conformal transformation, the visualizations and implementation of the ICP algorithm were performed within the Meshlab Software program (Pisa 2010).

5.3 Future Research

Future research into relating automatically relating Sparse Point Clouds to Faceted Models can take advantage of the fact that the user knows the position of every 3D point w.r.t. its derived imagery. This fact, combined with the ability to reorient a model to the same vantage point allows the projected image of that model to be registered to that same imagery (as

demonstrated in the next chapter). Figure 5-9 helps graphically display this critical concept to relate SPCs to FMs.

The secondary task is then to minimize the distance from the SPC points to the nearest model vertices, edges and/or planes. As a bi-product of this approach, the distance error from the model to the SPC can be utilized to indicate regions of the model that are inaccurate or that have changed since its creation. Due to the inherently diverse datasets, many solutions in this area will rely on nonlinear iterative techniques that seek to minimize error metrics both locally and globally.

It should be highlighted here, that feature matching via the use of facets utilizes geometric modeling at the facet level. That is, a facet is a geometric object that is at a higher level in the geometric hierarchy than points and lines. Something like a wall of a building is yet a higher level entity that could be a portion of a plane resolved from facets. So the general problem is to address techniques to enable the emergence of higher-level geometric entities from lower ones by a search and model building process that is supported by point clouds and imagery data. Of course, one question that will always remain is how a more complex model is "recognized" in a scene, given lower-level data and models (H. Rhody 2010).

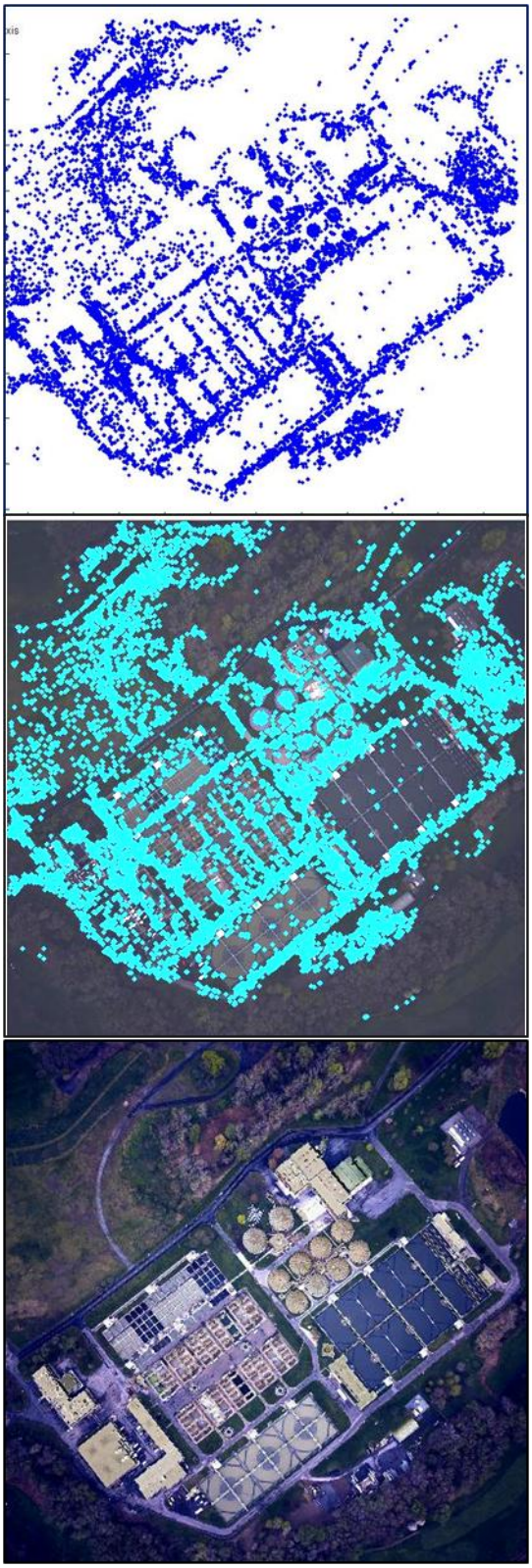


Figure 5-9 The Bundle Adjusted VanLare Site SPC (top), was projected back into the base image (Middle) and can then be compared directly with the FM where the base image is used as a UV texture on the terrain (Bottom).

5.4 Faceted Model to Faceted Model

Here we wish to relate two similar, and in some cases the same, 3D models of a scene. This will be necessary to relate models within various coordinate systems, both absolute and relative, as well as ones that may have been indirectly altered due to the effect of the modeling environment. For example, when a complex model with various components is brought into Google Earth, each component may be required to "settle" individually on the terrain. This can have the effect of changing the relative altitude of each component with respect to other elements of the modeled scene.

5.4.3 Approach

As with a few of the previous approaches, both automated and user assisted techniques are available to relate the models. However, the model datasets should be more similar, if not more reliable.

5.4.3.1 *Using Distance Similarity*

If there is any internal consistency to the model dimensions and relative structure, the 3D distance metric (similar to the 2D version in Section 2.2.1) can be utilized to automatically relate model vertices within prescribed error bounds. This could be applied in a regional sense to create similar feature matches in a localized area to mitigate the terrain "settling" effects mentioned earlier. In this way it would be possible to detect this terrain effect by comparing the relative mathematical relationship of individual building vertices compared to that of the mathematical model relating the whole scene.

5.4.3.2 *Using User Assisted Vertex Matching*

Although this approach is not completely automated, it could be made relatively painless by developing techniques such as a "stick cursor" that would automatically highlight the nearest vertex to the current cursor location. Only three "good" vertex correspondences are required to uniquely solve for the seven 3D conformal transform parameters (Scale, T_x, T_y, T_z, ω, ϕ, κ). Of course, as with most of these techniques, additional correspondences can be used to minimize the effects of noise and model imprecision.

5.4.4 Case Study

This study describes a situation where it was necessary to relate the FM of VanLare (Pictometry 2010) within the world coordinate system of Google Earth and the same model within the local coordinate system of the Advanced Analyst Exploitation Environment (AANEE). For this study, user assisted selection of 12 matching vertices (Figure 5-10) were used to develop a Conformal 3D transform (0), to relate the two models.

In addition to the previously mentioned terrain "settling" of the building in Google Earth, there is a limited ability to precisely pick global vertex coordinates. This was most notable in elevation, where the precision was limited to ~1m. The measurement error in the Latitude and Longitude is estimated at ~0.5m. Finally, due to the 30m terrain and unknown model placement accuracy, any coordinate transformation within ±15m error is probably within the measurement accuracy and acceptable error bounds of this case study. As seen in Table 4, good RMSDE results were obtained for most model correspondences, even considering the

measurement limitations and GE influences. It should be noted that the author enforced a unity scale factor when applying his Conformal 3D Transform since both models were the same.

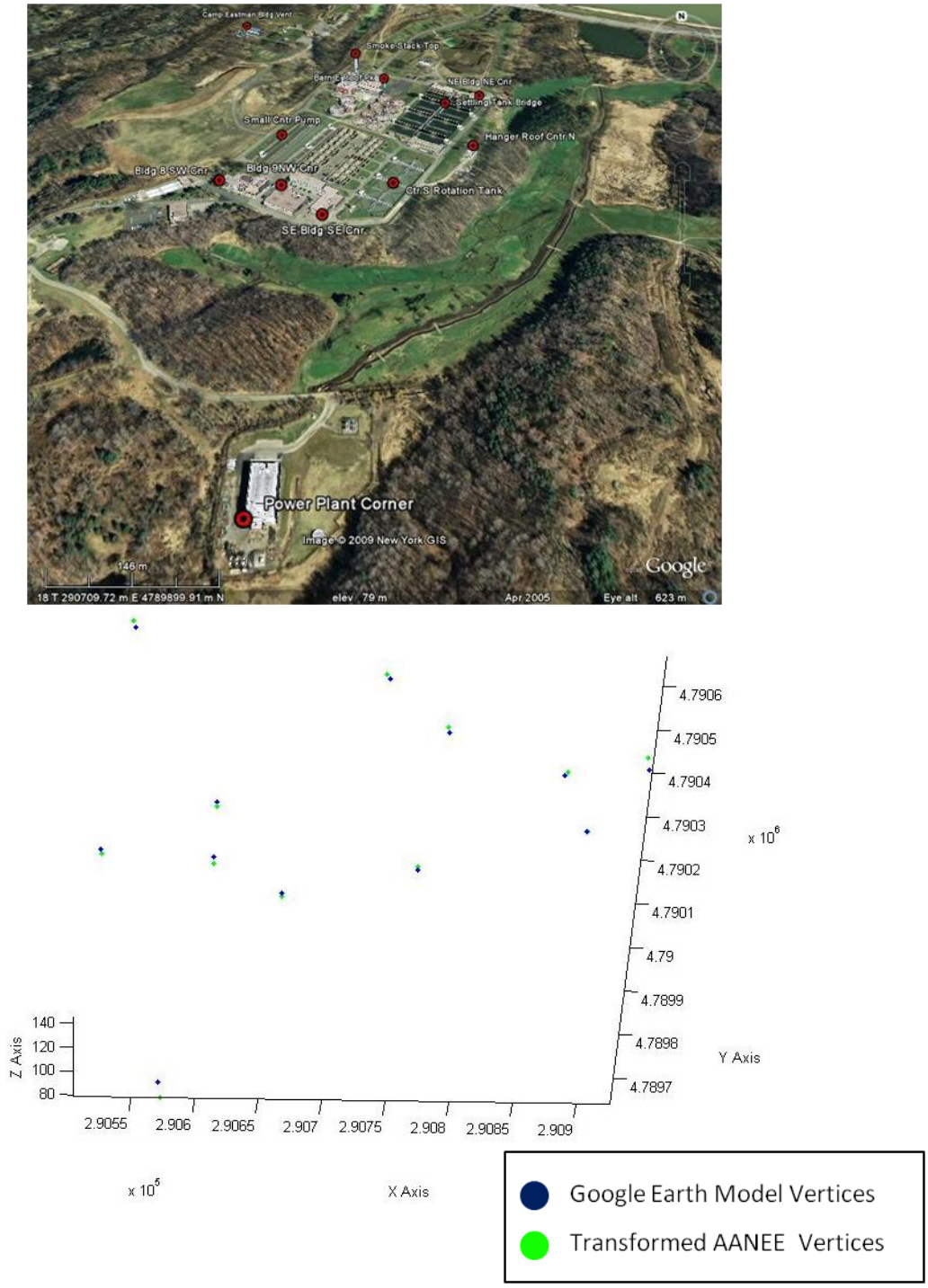


Figure 5-10 The Control Points used to related the GE and AANEE models (top) and the resulting transformation of the local points into Global UTM coordinates when compared to their matching Google Earth locations (bottom).

Table 4 The following table provides more explicit evidence of the actual transformation performance of Figure 5-10.

	Google Earth Coords [m]			AANEE Local Coords [m]			AANEE Global Coords [m]			
Control Pt Location	**X**	**Y**	**Z**	**x**	**y**	**z**	**X'**	**Y'**	**Z'**	**RMSDE [m]**
Settling Tank Cross	290861.0	4790341.6	97.0	294.7	-62.9	23.3	290863.8	4790336.8	95.6	3.3
Smoke Stack Top	290719.7	4790434.4	142.0	146.1	-19.2	-71.0	290718.9	4790434.5	146.0	2.3
Power Bldg	290571.7	4789639.8	92.0	23.7	-65.2	727.6	290572.8	4789639.0	82.0	5.8
Eastman Park Bldg	290509.2	4790666.6	97.0	-70.8	-63.1	-296.7	290508.2	4790667.8	109.2	7.1
Barn Peak	290767.6	4790414.7	104.5	194.5	-55.1	-51.4	290766.0	4790414.3	107.2	1.8
NE Bld NE Corner	290926.7	4790354.6	98.0	355.9	-58.8	2.5	290925.5	4790355.8	99.2	1.2
Hangar Roof N Peak	290885.1	4790195.0	103.8	319.6	-58.5	162.9	290884.7	4790196.2	96.6	4.2
Ctr S Rotation Tank	290754.9	4790119.7	97.0	191.6	-58.8	241.7	290754.6	4790120.8	96.7	0.7
SE Bldg SE Corner	290651.0	4790047.1	103.0	90.1	-55.6	316.3	290651.2	4790048.9	100.1	2.0
Bldg 9 NW Corner	290595.1	4790110.5	108.5	32.1	-51.9	255.6	290595.1	4790111.3	106.4	1.3
Bldg 8 SW Corner	290505.4	4790132.4	106.3	-57.2	-51.6	236.6	290506.5	4790132.8	108.8	1.6
Small Center Pump	290591.7	4790249.7	103.8	24.8	-56.4	119.3	290591.6	4790248.1	105.0	1.2
Total Ave RMSDE	Transformation: Conformal 3D Rigid Body									2.7

The following 3D Homography was created using the author's Conformal 3D algorithm using the matching points. It was then used to transform the AANEE model from its local coordinate system into a Global UTM coordinate system:

$$H = \begin{vmatrix} 0.9982 & -0.0288 & 0.0180 & 290571.3807 \\ -0.0283 & -1.0017 & -0.0243 & 4790366.9761 \\ -0.0179 & -0.0228 & 1.0432 & 166.9905 \\ 0.0000 & 0.0000 & 0.0000 & 1.0000 \end{vmatrix}$$

5.5 Constraining the Transform – 3D Conformal and Affine

As mentioned in Section 2.4, constraining the transform results can be a powerful tool for ensuring that minimal corruption occurs to the data during the relational process. This is important to ensure models retain their internally consistent dimensions and so that the

process intensive results of our SBA process keep their rigid body relationships. This data integrity issue is very similar to retaining radiometric accuracy during the resampling of images during the transformation process.

5.5.1 Conformal 3D Transform

Although 3D 'rigid body' transformations traditionally only include translation and rotation elements ($S = 1$), it will be referenced here in conjunction with the uniform scaling parameter common to the Conformal Transform; where $S = S_x = S_y = S_z$ or $S = \frac{(S_x + S_y + S_z)}{3}$. This will still preserve the internal geometry of the relative angles and distance ratios.

Conformal Transform

$$X = RSx + T \tag{67}$$

Conformal Sub-Matrices Transform

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} S & 0 & 0 \\ 0 & S & 0 \\ 0 & 0 & S \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{68}$$

3D Non-Homogeneous Transform

$$H_{3\times4} = [SRx \quad T] = \begin{bmatrix} SR_{11} & SR_{12} & SR_{13} \\ SR_{21} & SR_{22} & SR_{23} \\ SR_{31} & SR_{32} & SR_{33} \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \tag{69}$$

3D Rotation Matrix

$$R = R_\kappa R_\varphi R_\omega = \begin{bmatrix} c\kappa & -s\kappa & 0 \\ s\kappa & c\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\varphi & 0 & s\varphi \\ 0 & 1 & 0 \\ -s\varphi & 0 & c\varphi \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\omega & -s\omega \\ 0 & s\omega & c\omega \end{bmatrix} \tag{70}$$

3D Composite Homogeneous Transform

$$H_{4\times4} = \begin{bmatrix} Sc\varphi c\kappa & S(s\omega s\varphi c\kappa - c\omega s\kappa) & S(c\omega s\varphi c\kappa + s\omega s\kappa) & T_x \\ Sc\varphi s\kappa & S(s\omega s\varphi s\kappa + c\omega c\kappa) & S(c\omega s\varphi s\kappa - s\omega c\kappa) & T_y \\ -Ss\varphi & Ss\omega c\varphi & Sc\omega c\varphi & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{71}$$

3D Conformal Parameters & Control Points

$$\begin{bmatrix} X_1 & X_2 & \cdots X_i \\ Y_1 & Y_2 & \cdots Y_i \\ Z_1 & Z_1 & \cdots Z_i \\ 1 & 1 & \cdots 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} & H_{14} \\ H_{21} & H_{22} & H_{23} & H_{24} \\ H_{31} & H_{32} & H_{33} & H_{34} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots x_i \\ y_1 & y_2 & \cdots y_i \\ z_1 & z_2 & \cdots z_i \\ 1 & 1 & \cdots 1 \end{bmatrix} \tag{72}$$

The three translation parameters can be easily extracted from the 4th column of H, and the 3 rotation and scale parameters can be obtained by utilizing the following equations ((73)-(76)). In (73), we utilize the property that the sum of the squares of the rows or columns of the rotation matrix must equal unity (DeWitt and Wolf 2000).

Extract Scale

$$S = \sqrt{\left(R_{11}^2 + R_{12}^2 + R_{13}^2\right)^{-1}} \quad \textbf{(73)}$$

Extract R_y

$$\varphi = sin^{-1}\left(\frac{R_{13}}{S}\right) \quad \textbf{(74)}$$

Extract R_x

$$\omega = sin^{-1}\left(\frac{R_{23}}{Scos\varphi}\right) \quad \textbf{(75)}$$

Extract R_z

$$\kappa = sin^{-1}\left(\frac{R_{12}}{Scos\varphi}\right) \quad \textbf{(76)}$$

Here, R is the 3x3 rotation matrix containing elements of the rotation about each of the axis ($R_x = \varphi$, $R_y = \omega$, and $R_z = \kappa$), which are often referred to as “roll”, “pitch”, and “yaw” when in reference to airborne platform motion. Additionally, care must be taken to avoid division by zero throughout many of these solutions, however since it is possible to test for this scenario it can often be avoided.

Equations (67) through (72) should clearly demonstrate the construction of the Conformal 3D Transform and how it can be applied to image correspondences to relate volumetric datasets. But, how can we derive the coefficients in such a way that constrains the results. The following technique was adapted from a 3D Pose Estimation algorithm (Haralick, et al. 1989) and modified to extract the scaling parameters for use in the 3D Conformal and Affine Transformations. The basic process is outlined below:

1. Determine location of both model centers
2. Translate model centers to origin (Demean Models)
3. Utilize point correspondences and Least Squares to derive the transform (H_{3x3})
4. Extract component Rotation and Scale Matrices using SVD and/or QR Decomposition
5. Rotate and Scale the original center coordinate of the working model
6. Translation is the base model center subtracted from the transformed model center

Once the models have been demeaned, the Translation parameters can be temporarily ignored and (67) can be simplified to the following:

Demeaned Models Transform

$$X = RSx \tag{77}$$

Scale & Rotation Matrices

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix} \begin{bmatrix} S & 0 & 0 \\ 0 & S & 0 \\ 0 & 0 & S \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{78}$$

Scale and Rotation Transform

$$H_{3\times3} = [SR] = \begin{bmatrix} SR_{11} & SR_{12} & SR_{13} \\ SR_{21} & SR_{22} & SR_{23} \\ SR_{31} & SR_{32} & SR_{33} \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \tag{79}$$

Parameters & Control Points

$$\begin{bmatrix} X_1 & X_2 & \cdots X_i \\ Y_1 & Y_2 & \cdots Y_i \\ Z_1 & Z_1 & \cdots Z_i \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} \\ H_{21} & H_{22} & H_{23} \\ H_{31} & H_{32} & H_{33} \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots x_i \\ y_1 & y_2 & \cdots y_i \\ z_1 & z_2 & \cdots z_i \end{bmatrix} \tag{80}$$

Pseudo-Inv Solution

$$H = Xx^{\dagger} \tag{81}$$

Singular Value Decomposition

$$[V, D, U] = SVD(H) \tag{82}$$

Where D is a diagonal matrix containing the singular values and V and U are unitary matrices such that $\mathrm{H} = \mathrm{V} * \mathrm{D} * \mathrm{U}$. Using the resulting decomposition of (82), allows us to retrieve the Rotation from the following relationship.

$$R = V\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & \det(VU^T)\end{bmatrix}U^T \quad \textbf{(83)}$$

This gives a unique solution for the rotation matrix R provided the rank of $H > 1$ and $\det(VU^T) = 1$ or rank of $H > 1$ and the minimum singular value is a simple root. This can be easily tested for to ensure the integrity of the rotation matrix.

Since the initial pose estimation technique did not require utilizing the singular values embedded within the D matrix for anything other than R solution validation, it wasn't utilized further. However, the author noticed the embedded scaling parameters when utilizing synthetic data and recalled decomposing the camera projection matrix into the IOPs and EOPs using QR Decomposition. Recall that the main diagonal of the IOP matrix (K), is related to the pixel pitch and focal length which provides scaling/magnification. In this way it is possible to utilize QR Decomposition directly to extract the Scale and Rotation parameters if care is taken concerning the sign of the retrieved values (negative scale parameters must be injected back into the rotation elements).

Although it is possible to utilize SVD to extract the Scale parameters, since they are related to the singular values through the geometric interpretation of the SVD. These axes are orthogonal eigenvalues and ranked from largest to smallest as in principle component analysis, which is probably not the diagonal scale component (S_x, S_y, S_z) ordering required for direct scale use. However, the proper scale components (S_x, S_y, S_z) can be determined using the following equations (84)-(86) or by using QR Decomposition as noted above.

Scale X-axis

$$S_x = \sqrt{\left(R_{11}^2 + R_{21}^2 + R_{31}^2\right)^{-1}} \quad \textbf{(84)}$$

Scale Y-axis

$$S_y = \sqrt{\left(R_{12}^2 + R_{22}^2 + R_{32}^2\right)^{-1}} \quad \textbf{(85)}$$

Scale Z-axis

$$S_z = \sqrt{\left(R_{13}^2 + R_{23}^2 + R_{33}^2\right)^{-1}} \quad \textbf{(86)}$$

For use with the 3D Conformal Transformation the x, y and z scaling parameters can be averaged for uniform scaling application as an initial linear estimate. In Case Study 5.4.4, since both models were from the same source they had the same scale ($S = 1$) and so the identity matrix could be utilized directly in place of the derived scale parameters.

Finally, the Translation is derived by subtracting the base model center from the transformed model center using (87),

Translation

$$T = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \bar{X} - RS\bar{x} \quad \textbf{(87)}$$

where $\bar{X}$ is the base model centroid and $\bar{x}$ is the transformed model centroid.

By using this process, it is possible to overcome much of the difficulties normally associated with developing a good linear estimate of the 3D relationship, even with the nonlinear interaction of the parameters associated with these types of problems in photogrammetry, geodesy, and remote sensing.

5.5.2 Affine 3D Transform

As with the 2D Affine Transformation, the 3D Affine Transformation includes translation (T), rotation (R), scale (S), and shear (W).

3D Affine Transform

$$\vec{X} = RSW\vec{x} + T \tag{88}$$

However, each of these now contain more parameters due to the additional dimensionality of volumetric space. Here the uniform scale will be replaced with three independent scale components (S_x, S_y, and S_z), which take the following matrix form.

NonUniform Scale Matrix

$$S = \begin{bmatrix} S_x & 0 & 0 \\ 0 & S_y & 0 \\ 0 & 0 & \mathrm{S_z} \end{bmatrix} \tag{89}$$

Although the full 3D Affine does not preserve internal angles due to the possible effects of shear (Eqs. (90) to (96)); it does preserve the parallelism of lines and planes.

X Shear

$$X = x + Sh_{xy}y + Sh_{xz}z \tag{90}$$

Sh_x Matrix

$$W_x = \begin{bmatrix} 1 & 0 & 0 & 0 \\ Sh_{xy} & 1 & 0 & 0 \\ Sh_{xz} & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{91}$$

Y Shear

$$Y = Sh_{yx}y + y + Sh_{yz}z \tag{92}$$

Sh_y Matrix

$$W_y = \begin{bmatrix} 1 & Sh_{yx} & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & Sh_{yz} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{93}$$

Z Shear

$$Z = Sh_{zx}y + Sh_{zy}z + z \tag{94}$$

Sh_z Matrix

$$W_z = \begin{bmatrix} 1 & 0 & Sh_{zx} & 0 \\ 0 & 1 & Sh_{zy} & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{95}$$

3D Shear Transform

$$\vec{X} = W\vec{x} \tag{96}$$

3D Shear with Control Points

$$\begin{bmatrix} X_1 & X_2 & \cdots X_i \\ Y_1 & Y_2 & \cdots Y_i \\ Z_1 & Z_1 & \cdots Z_i \\ 1 & 1 & \cdots 1 \end{bmatrix} = \begin{bmatrix} 1 & Sh_{yx} & Sh_{zx} & 0 \\ Sh_{xy} & 1 & Sh_{zy} & 0 \\ Sh_{xz} & Sh_{yz} & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots x_i \\ y_1 & y_2 & \cdots y_i \\ z_1 & z_2 & \cdots z_i \\ 1 & 1 & \cdots 1 \end{bmatrix} \quad \textbf{(97)}$$

The same approach utilized for the Conformal 3D Transform will be utilized for the Affine since it supports the three independent scale parameters and since the shear component will often be assumed as negligible for our applications.

5.5.3 Homogeneous 15 Parameter Linear Estimate

Of course the simplest way to get a 3D estimate is to utilize the unconstrained 15 parameter homogenous approach and then utilize the nonlinear minimization and weighting technique of the next section (5.5.4) to narrow in on the correct solution. The equations below are the 3D incarnation of the 2D approach covered in Section 2.3.

$$\begin{bmatrix} X_1 & X_2 & \cdots X_i \\ Y_1 & Y_2 & \cdots Y_i \\ Z_1 & Z_1 & \cdots Z_i \\ 1 & 1 & \cdots 1 \end{bmatrix} = \begin{bmatrix} H_{11} & H_{12} & H_{13} & H_{14} \\ H_{21} & H_{22} & H_{23} & H_{24} \\ H_{31} & H_{32} & H_{33} & H_{34} \\ H_{41} & H_{42} & H_{43} & 1 \end{bmatrix} \begin{bmatrix} x_1 & x_2 & \cdots x_i \\ y_1 & y_2 & \cdots y_i \\ z_1 & z_2 & \cdots z_i \\ 1 & 1 & \cdots 1 \end{bmatrix} \quad \textbf{(98)}$$

$$H = Xx^{\dagger} \quad \textbf{(99)}$$

It is important to note that this linear approach solves for 15 DOF and requires 5 non-collinear 3D point correspondences. This is obviously many more than just the 7 parameters required for a conformal 3D relationship and so may induce undesired, higher order effects like projection and skew. However, the linear solution must only obtain an estimate within the capture region of the global minimum through the use of a nonlinear solver such as LMA and provide a reasonable starting point to minimize against the desired parameters.

5.5.4 Nonlinear Minimization and Weighting

Since the 3D Conformal transform is nonlinear in its solution for the scale and rotation parameters, it is necessary to implement a nonlinear optimization method such as LMA to accurately solve. Once the initial estimate for the rotation angles, translation, and scale have been accomplished, those same parameters can be prepared to provide a nonlinear minimization. This process is very similar to Section 13.1, except that the cost function minimization is compared against the total squared 3D distance error, as opposed to the projected 2D distance error.

$$\sum_{i=1}^{n} \|X_i - x_i(R, S, t, X_M)\|^2 \quad \textbf{(100)}$$

Similar to the technique utilized in 5.5.1, the H_{4x4} matrix can be decomposed into the Translation (T), Scale (S) and Rotation (R) matrices to obtain the parameter estimates from the coefficients.

So, how do we optimize for a solution that is only dependant on the desired 7 Conformal Transform parameters? A useful technique to accomplish this is to start with the results of the 4x4 Homography; it is then possible to induce a weighting function w on the undesired terms and increase it at every iteration of the cost function computation.

$$\sum_{i=1}^{n} \|X_i - x_i(R, S, t, X_M)\|^2 + w(S_x - S_y - S_z)^2 + wSh_x{}^2 + wSh_y{}^2 + wSh_z{}^2 \quad \textbf{(101)}$$

This has the effect of slowly pulling the solution toward the desired constraints, where $Scale = S_x = S_y = S_z$ and $Shear = Sh_x = Sh_y = Sh_z = 0$. Once they are within a certain threshold of these constraints, they can be clamped to their desired values for a final

estimation (Hartley and Zisserman 2004). The LMA implementation is otherwise very similar to that utilized in Chapter 13. Although this approach was not implemented by the author, it has been retained in this document for completeness and due to its general application in reducing the solution space of results, which is often a key aspect of registration accuracy.

In the next chapter, most of the techniques developed in the preceding chapters will be utilized to enable the challenging area of multimodal registration. Here physical modeling of scene materials will be implemented to augment the 3D site model. This will allow for simulations in various modalities of interest while maintaining the proper scene appearance, which is critical for automated registration.

6 Multimodal 3D Registration

In the previous sections, essential techniques have been developed to relate images within a 3D construct, allowing for robust mathematical relationships between the datasets of interest even in the presence of parallax and occlusion. In this section, we will utilize a 3D "model-centric" environment to compensate for the viewing parameters of the sensors at the time of image acquisition. This will allow us to model and mitigate the effects of terrain/building relief, shadowing effects and occlusion. At the same time, we will utilize the ability of a physics based simulator, the Digital Imaging and Remote Sensing Image Generator (DIRSIG) program, to estimate the appearance of various modalities under different lighting/atmospheric conditions and sensor parameters (Section 6.3), to produce representative simulated imagery (Figure 6-1).

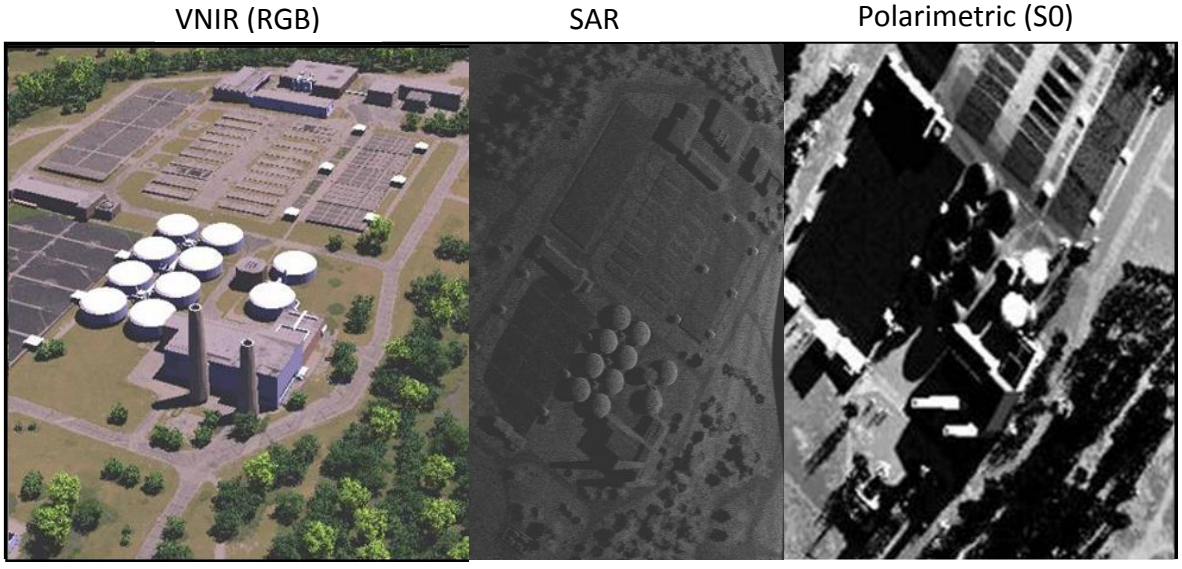


Figure 6-1 Multimodal image synthesis using DIRSIG's physics based modeling [courtesy Dr. Mike Gartley (Gartley, et al. 2010)].

If modeled properly, this has the potential to relate even the most diverse datasets, such as polarimetric, thermal and Synthetic Aperture RADAR (SAR) imagery. By using an inherently 3D

approach to address the viewing geometry effects, coupled with DIRSIG to address the physical appearance of the scene, robust Multimodal 3D Registration is possible.

The images shown below in Figure 6-2, displayed within Google Earth, visibly show the inverted contrast of both water (circled) and vegetation when imaging the same site in the visible and infrared regions of the spectrum. This can frustrate correlation and feature based registration techniques. However, the edge detail can often still be utilized for common feature generation. Also, techniques like Maximum Mutual Information have been demonstrated to successfully relate multimodal imagery (Fan, Rhody and Saber 2008) once the 3D influences are removed.

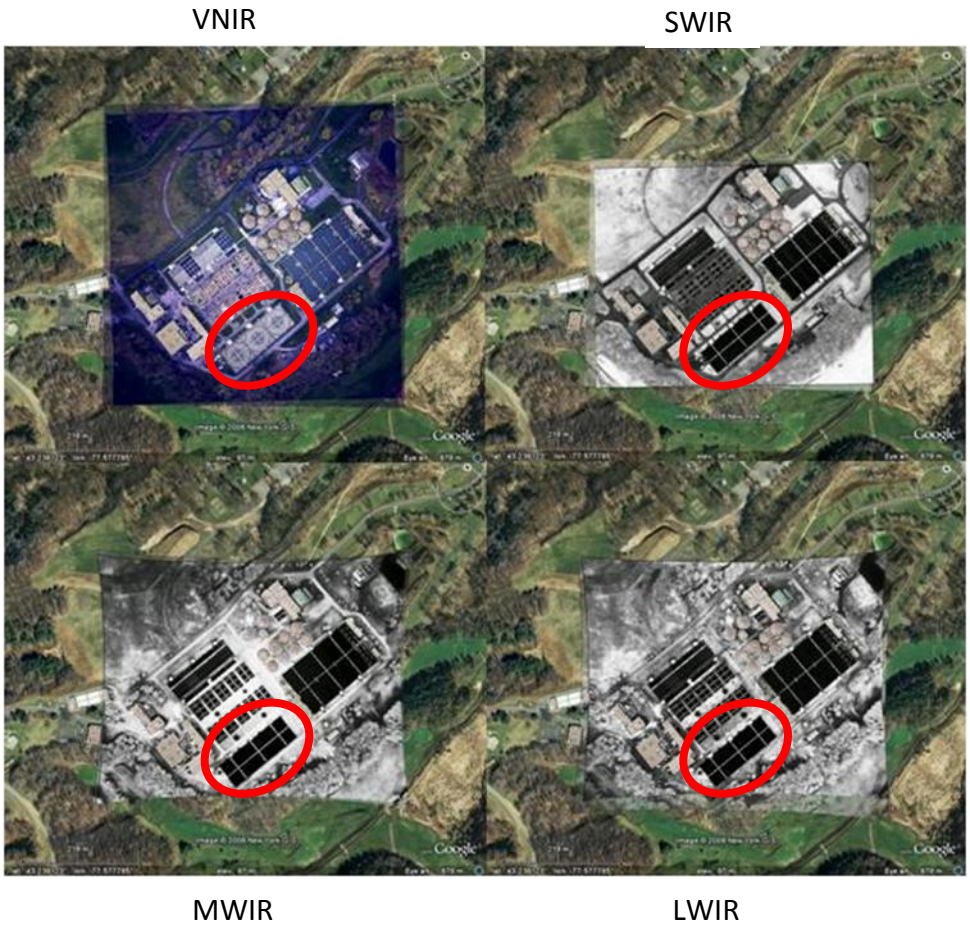


Figure 6-2 Multimodal imagery registered to GE textured terrain using user assisted GCP selection and overlaid upon the initial sensor derived (IMU/GPS) global coordinate predictions. The inverted contrast of water in VNIR and Infrared is circled.

6.1 The 'Model Centric' Approach

By orienting a site model to the pose of the sensor it is possible to mitigate the 3D projective effects of the scene view compared to the image (Walli and Rhody, Automated Image Registration to 3-D Scene Models 2008). Once this is accomplished, a physics based simulation of the scene is rendered in order to estimate its modality specific appearance. This 'model centric' approach has the potential to mitigate even the most challenging issues of parallax, occlusions, shadowing, and diverse multimodal appearance (Van Nevel 2001), which currently plague automated registration of diverse views imaged from across the electro-magnetic spectrum (Figure 1-2).

However, once the image and model projection are accurately registered, the process is not complete until the real image is then mapped back to the site model in order to regain the depth information that was lost when the image was acquired. The entire modeling, simulation, mathematical relationship and archival (MSRA) process can be visualized in Figure 6-3. A key thrust here, is an understanding that 3D multimodal registration involves image to model registration and archival. Once the model and image have been properly related, it is possible to archive the image as a texture onto the mode as a database of "layers".

MSRA Approach

A) Model

B) Simulate

C) Relate

D) Archive

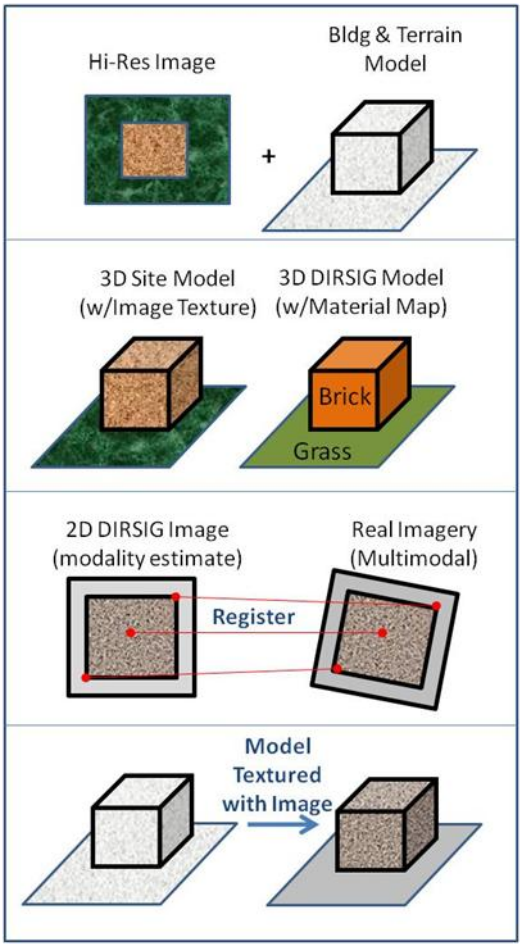


Figure 6-3 This figure illustrates the MSRA Approach to 3D Multimodal Registration, where A) is the modeling phase, B) is the physics based simulation phase, C) is the 2D image registration phase, and D) is the Image archival phase onto a model.

6.2 Model - Geometrically

Here, the geometric modeling step will be broken down into 3 separate flavors: Existing/User Defined, LIDAR Derived, and Multiview Image Derived. These modeling paths have been further defined into levels of fidelity which relate to increasing degrees of realism/complexity within the models (Figure 6-4).

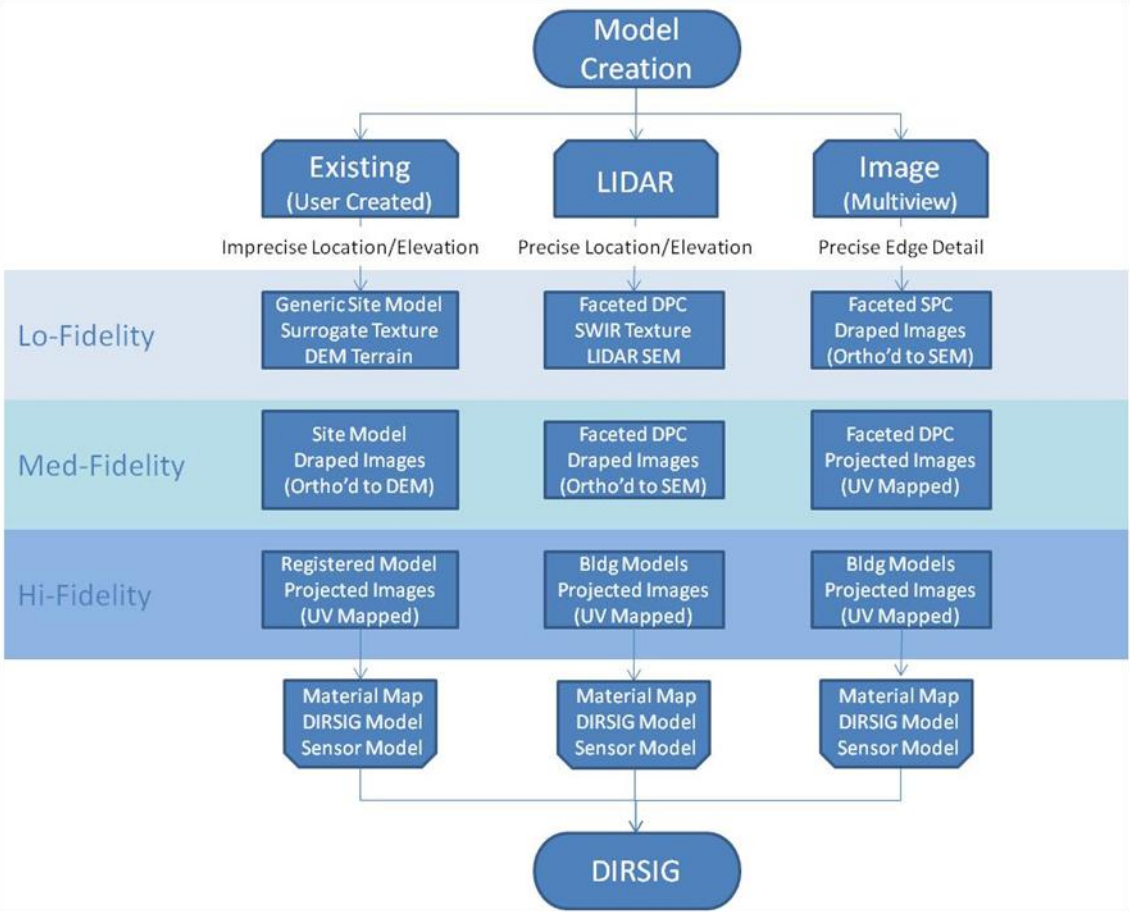


Figure 6-4 This flowchart illustrates three different paths for generating geometric models for DIRSIG simulation. From left-to-right they are Existing/User Created, LIDAR Derived, and Multiview Image Derived models with varying degrees of fidelity.

6.2.1 Existing/User Created Model

Some of the more realistic geometric models that already exist for a site will be textured with imagery or photographs, such as the model shown below (Figure 6-5), courtesy Pictometry International (Pictometry 2010).

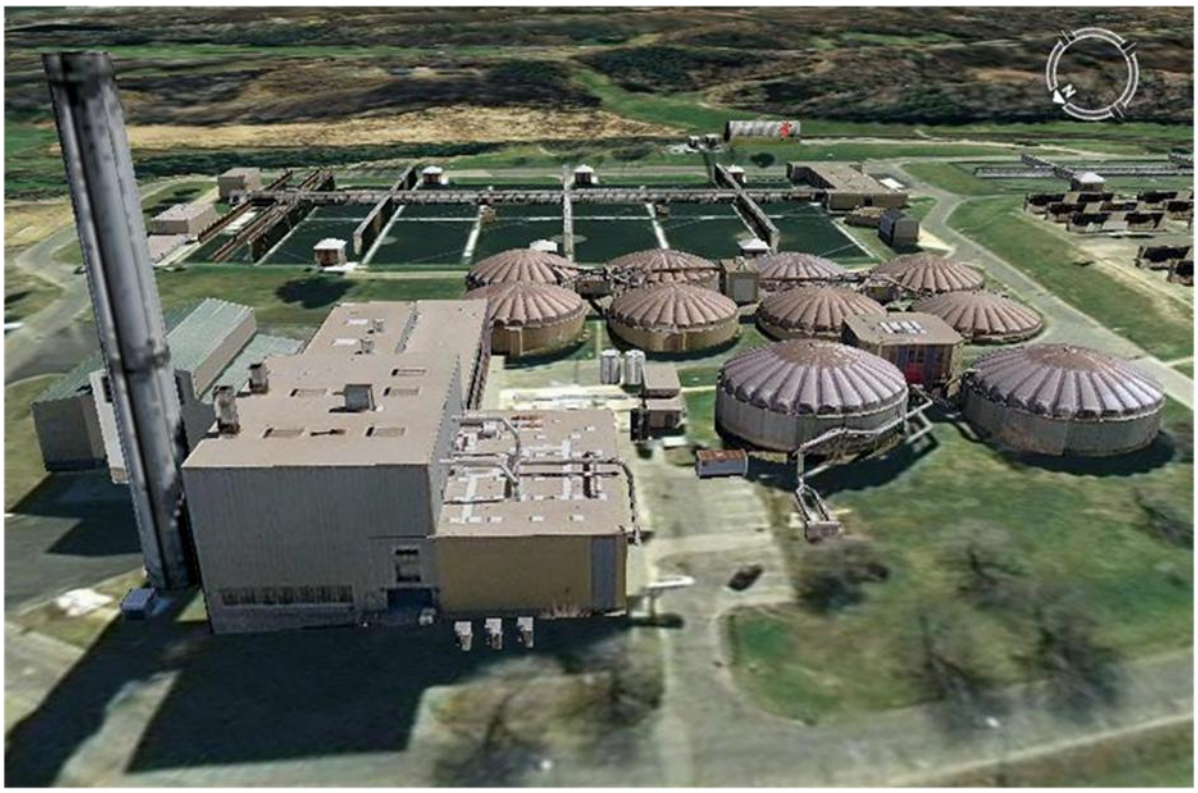


Figure 6-5 This Hi-Fidelity model of the VanLare Waste Water Processing plant is representative of an existing geometric model placed in Google Earth that utilizes UV mapped image textures for added realism (courtesy Pictometry Int.)

Additionally, Figure 6-6 shows the basic modeling approach for utilizing an existing 3D model as the underlying geometry for a DIRSIG simulation. Here, specific facets of the model are attributed with real material spectra using field data collected by an Advanced Spectral Devise (ASD), but, alternatively one could use Hyper Spectral (HS) data collected from an airborne sensor. This spectral information is used to physically estimate how a simulated target material should look when viewed from sensors imaging in various spectral ranges and modalities.

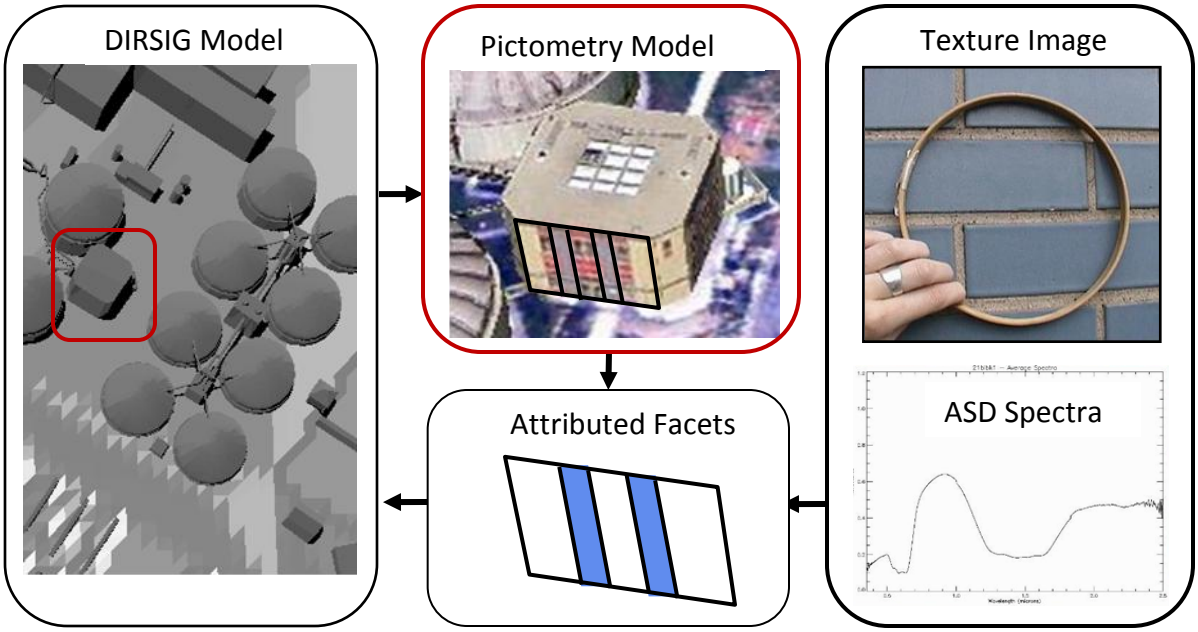


Figure 6-6 This illustration depicts the process of adding spectral reflectance curves to a realistic scene model in DIRSIG using Hyperspectral or Advanced Spectrometer Data (ASD) to properly simulate material appearance in various spectra.

Alternatively to the traditional technique of mapping specific facets with material spectra within DIRSIG, it is possible to utilize a texture mapped in the $uv\ plane$ (UV Texture Map) to the unwrapped model (Figure 6-7). This process flattens (unwraps) the model into a 2D representation that allows direct association of the model vertex locations with that of an image mapped to a normalized $uv\ plane$. It should be noted that this image is often a composite of several images pieces that relate directly to model facets such as walls and roofs.

Not only does this type of texture add realism to the geometric model, it allows for oblique imagery of a scene to be related to the DIRSIG model, resulting in the sides of buildings displaying representative features. To use the UV Texture approach within DIRSIG, it is necessary to associate the uv-mapped texture image with a grayscale lookup table (LUT) to relate the image textures to specific material spectra. This was accomplished by first

generating a K-Means segmentation of the texture image within the ENVI Software program, under the 'Unsupervised Classification' algorithms (ITT Visual Information Solutions 2008). Depending on the results of the automated K-Means clustering technique, some user-assisted segmentation may be required to clearly define visible material boundaries (Figure 6-8).

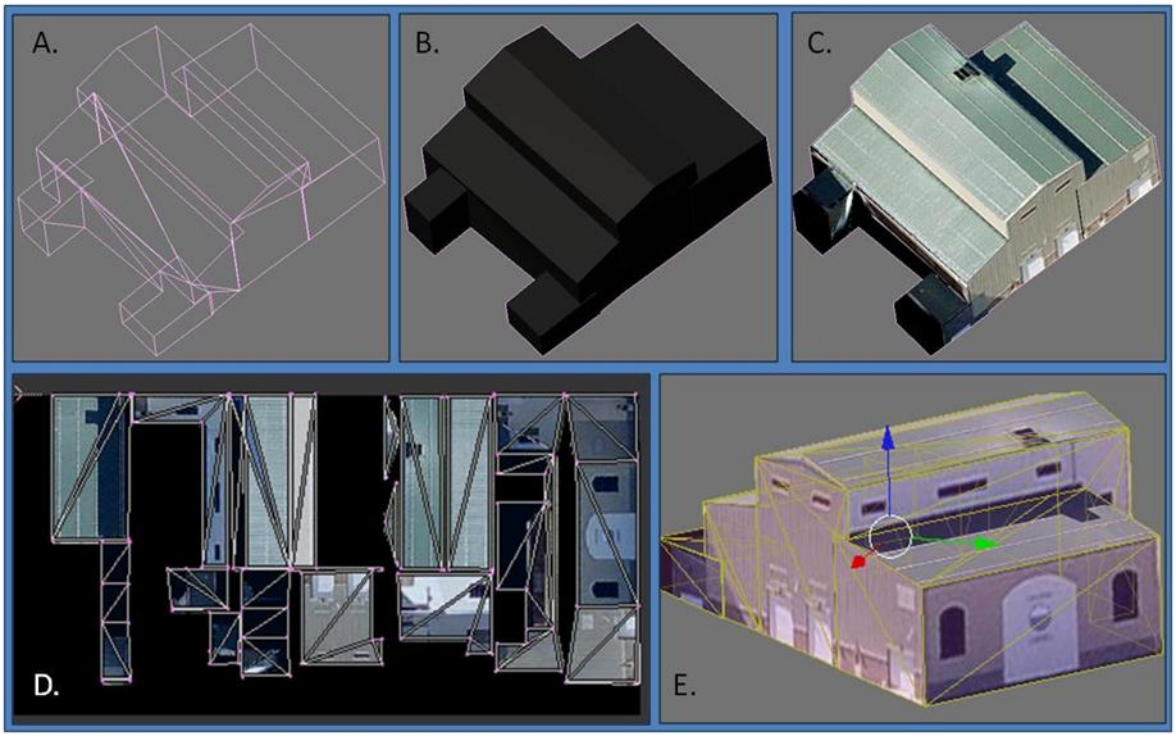


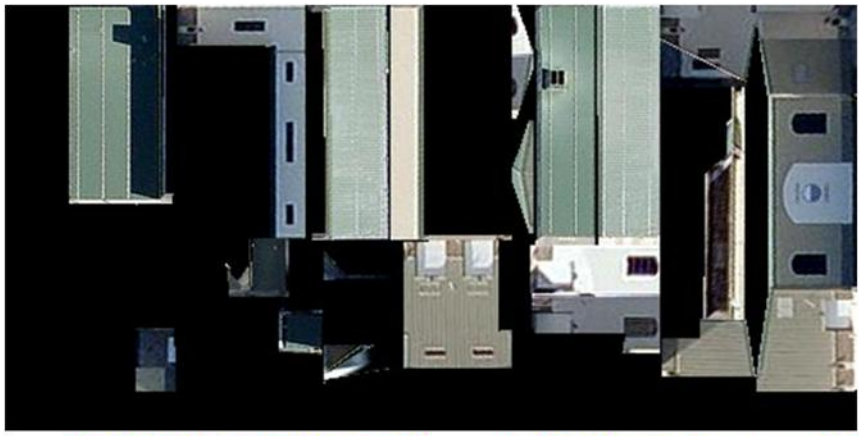
Figure 6-7 Illustrates the UV Texturing process: A) The wireframe model, B) The faceted model, C) The UV textured Model, D) The flattened (unwrapped) model with overlaying image texture, and E) The textured wireframe model.

The resulting material class-map image can then be associated with the texture-map image within DIRSIG to add both material identification and spatial texture characteristics to regions within a given model or model facet. For additional details on incorporating UV textured models into DIRSIG, reference Appendix E in Section 15.2 and Appendix F.

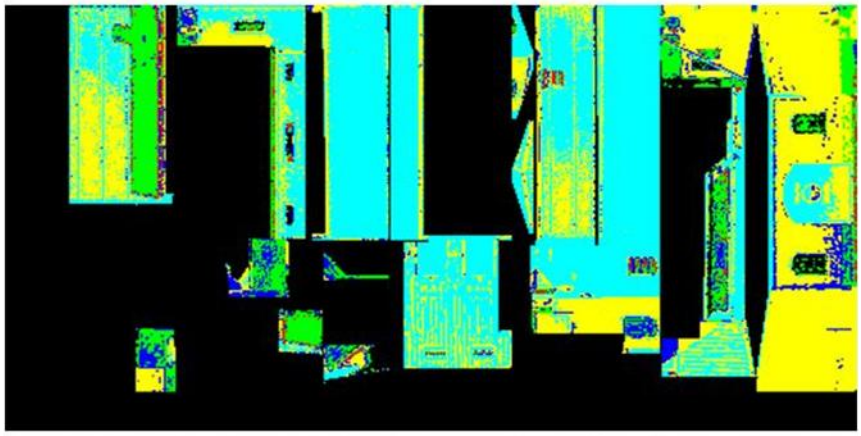
Unless additional information is available to augment the model creation (as in the following sections), the terrain will most likely be relegated to DTED Elevation Map (DEM) quality; this

equates to ~30m posting for most areas on the globe. While this fidelity of terrain is good enough to perceive major topological influences, such as mountains, valleys, and bodies of water; it is normally not detailed enough to detect building placements, roads, and minor

A. Texture Map



B. Class Map (K-Means)



C. Material Map

Material LUT

DC	Mat ID	Description
0	0	Black
136	7129	Glass
187	1012	Tan Mtl
224	1021	Green Mtl
255	1004	White Mtl

Figure 6-8 This graphic illustrates the process used to turn a UV Texture map (A), into a material class map LUT (C) by first segmenting the image with a K-Means classifier (B).

terrain features. Additionally, modeling the 3D influences of trees will probably be constrained

to the synthetic generation of generic tree models placed using the help of overhead imagery. While this may be acceptable for many simulation activities, it will generally not be good enough for image registration, since it is necessary to have accurate bio-mass placement to accurately remove the 3D effects of viewing geometry on the scene. The utility of have some 3D information available, will be examined in the following two subsections (6.2.2 & 6.2.3).

6.2.2 Hybrid Models - Developing LIDAR Augmented models in DIRSIG

In this section, we will utilize various types of remotely sensed data to create a hi-fidelity geometric and physical model of a site for use within DIRSIG. The building models are imagery-derived, but, hand-made (Pictometry 2010); while the terrain and trees were derived from LIDAR data, and the terrain texture is from CITIPIX imagery (Kodak Global Imaging 2008). The general process utilized to create a hi-fidelity hybrid simulation in DIRSIG is flowcharted in Figure 6-9.

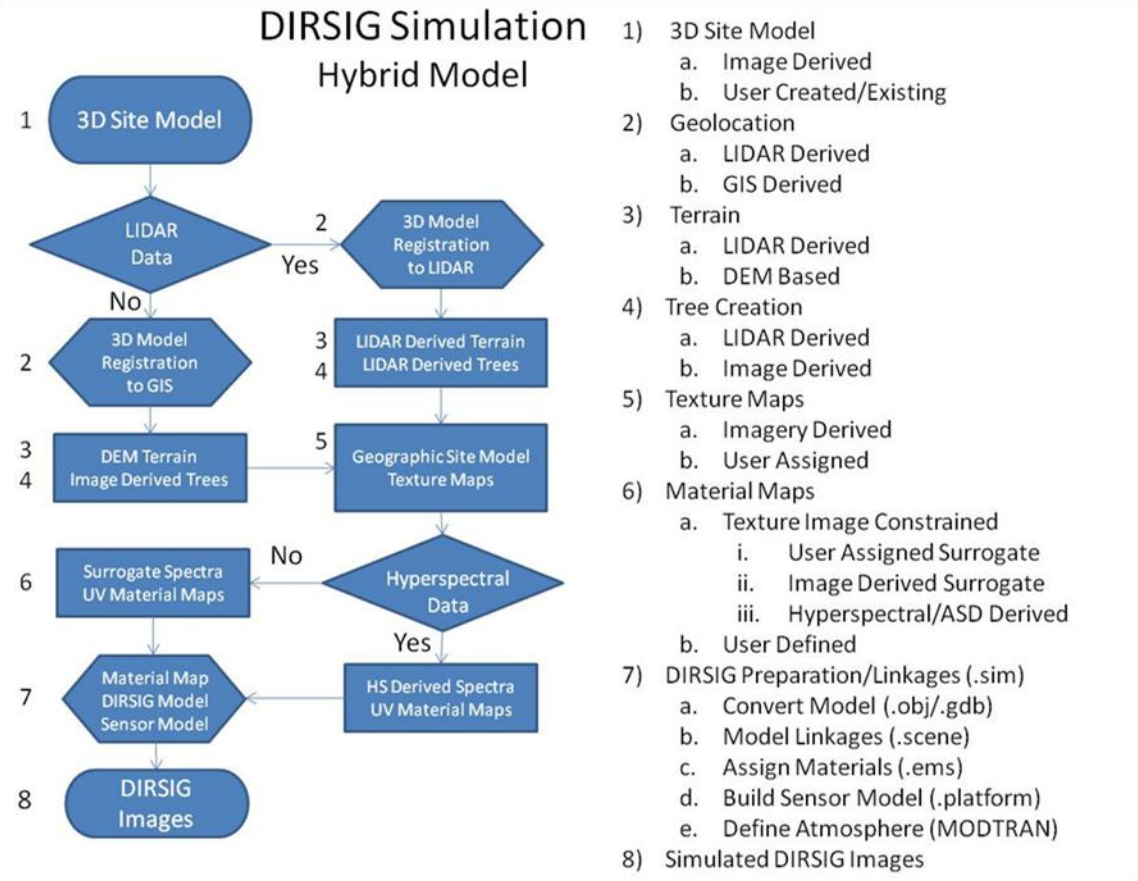


Figure 6-9 This flowchart depicts the process utilized for DIRSIG model creation using hybrid models and imagery.

The first step in developing a 3D model of a real site of interest should entail the use of LIDAR data, if it is available. The 3D positional information (Latitude, Longitude, & Altitude) that is available in the LIDAR range information is critical in developing accurate terrain, building placement, and elevation characteristics of the scene. In fact, even if a site has been accurately modeled by hand, using real imagery as a template (Section 5.2), the LIDAR data can be used as an anchor to ensure the model has accurate dimensions and geographic placement.

If the 3D model can be registered to the LIDAR data (using techniques such as in Section 5.2), its accuracy can be assessed through visual inspection or 3D change detection techniques. In this way, it is possible to utilize the inherent 3D nature of the LIDAR data synergistically with the detail rich edge information of the image derived building models. Figure 6-10 shows how this

process can be utilized to create an accurate hybrid site model that has hi-fidelity LIDAR derived terrain (~1 [m] postings), registered to the multiview image models from the previous section.

At this point, the hybrid model will contain the modeled buildings and terrain, but, none of the surrounding foliage. For accurate scene simulations, this may be important to simulate site obscuration and 3D canopy influences for accurate registration. Although significant research is being done in the area of tree identification using LIDAR data (Kim, Hinckley and Briggs 2009), which could be used to grow representative tree types at the correct locations, this often requires two LIDAR collections utilizing the leaf-on and leaf-off structural characterization. Additionally, the tree models would need to prescribe to the bio-mass restrictions defined by the LIDAR collection and although feasible, falls beyond the current scope of this research.

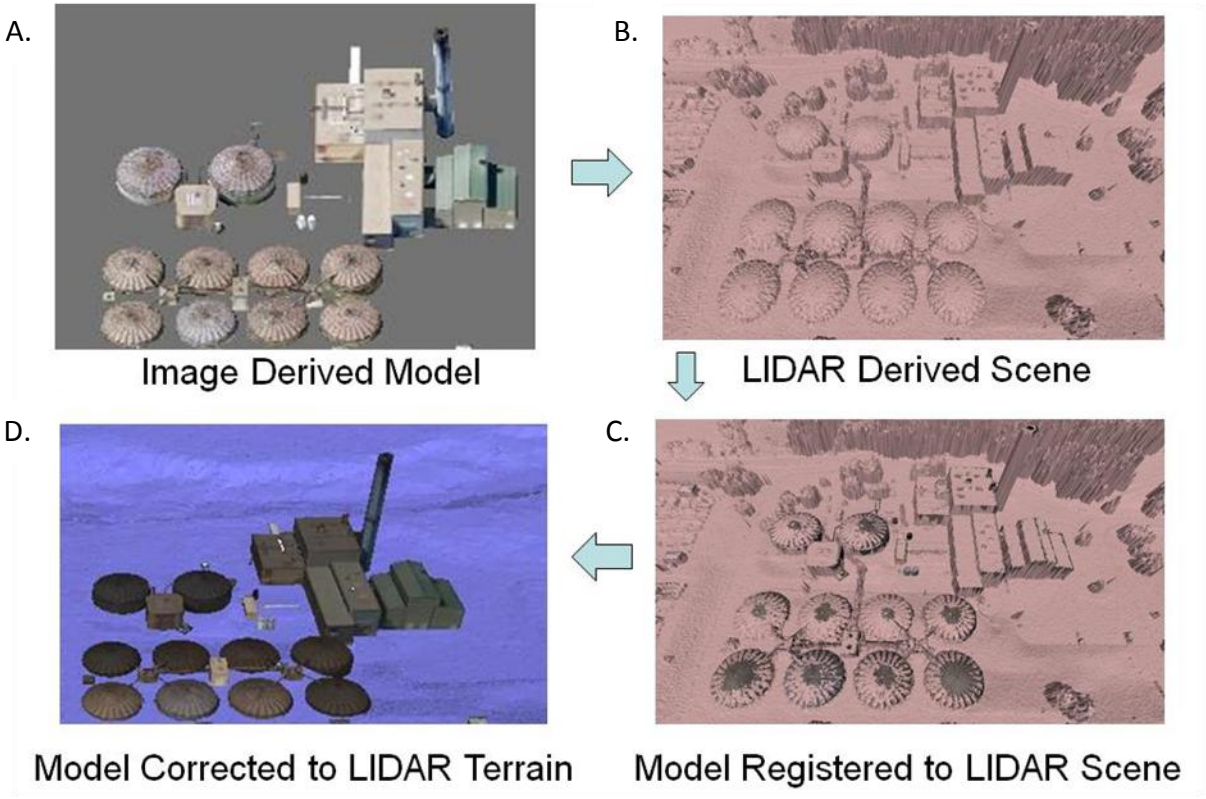


Figure 6-10 This figure illustrates the process utilized to register a site model (A), to a faceted LIDAR dataset (B), to assess model fidelity and to ensure proper building placement and dimensions (C). Finally the model is placed on the bare earth LIDAR terrain (D) to create a hybrid scene using both the LIDAR terrain and Image derived building models.

To approximate the biomass influences of a scene's foliage, a more straightforward approach was developed and implemented by the author. This approach utilizes the LIDAR information directly by placing a model facet at the location of each return that is 3m above the terrain, after removing the building returns. While many possible geometries could be utilized to represent the foliage 3D character (i.e. boxes, pyramids, or draped wings - Figure 6-11), each shape has potential benefits and detractors depending on the imaging scenario that is being simulated. Of primary concern is whether a downward-looking NADIR view or a more side-looking Oblique view is desired. For near-NADIR imaging situations, the flat square panel would represent the simplest basic shape, while still approximating the basic view acquired by most

sensors. Additionally, it still allows for the 3D influence of parallax to be modeled properly (Figure 6-11a), while allowing for foliage 'poke through' imaging of the scene. This would of course break down as the angular view to the scene approaches more oblique angles and would necessitate the examination/use of one of the other geometries. It should be noted that code was generated to automatically convert LIDAR returns into the basic shapes defined in Figure 6-11a (courtesy Niek Sanders) and Figure 6-11b due to their low facet count (each has 2 facets with 3 vertices/facet due to the triangulation requirements of most model entities).

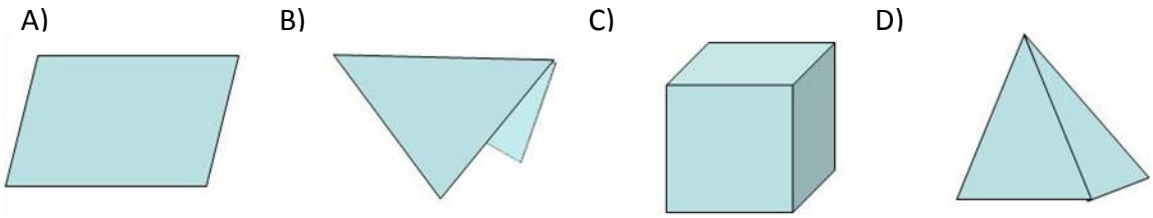


Figure 6-11 Example geometric shapes that could be used to represent tree foliage when paired with LIDAR point returns.

For the reasons stated above, the basic square model facet was chosen and placed at the location of each return, with the normal of these facets pointing straight up. This basic shape allows for a 'terrain-like' draped texture over every facet within DIRSIG; which, for near-NADIR imaging simulations is adequate for registration.

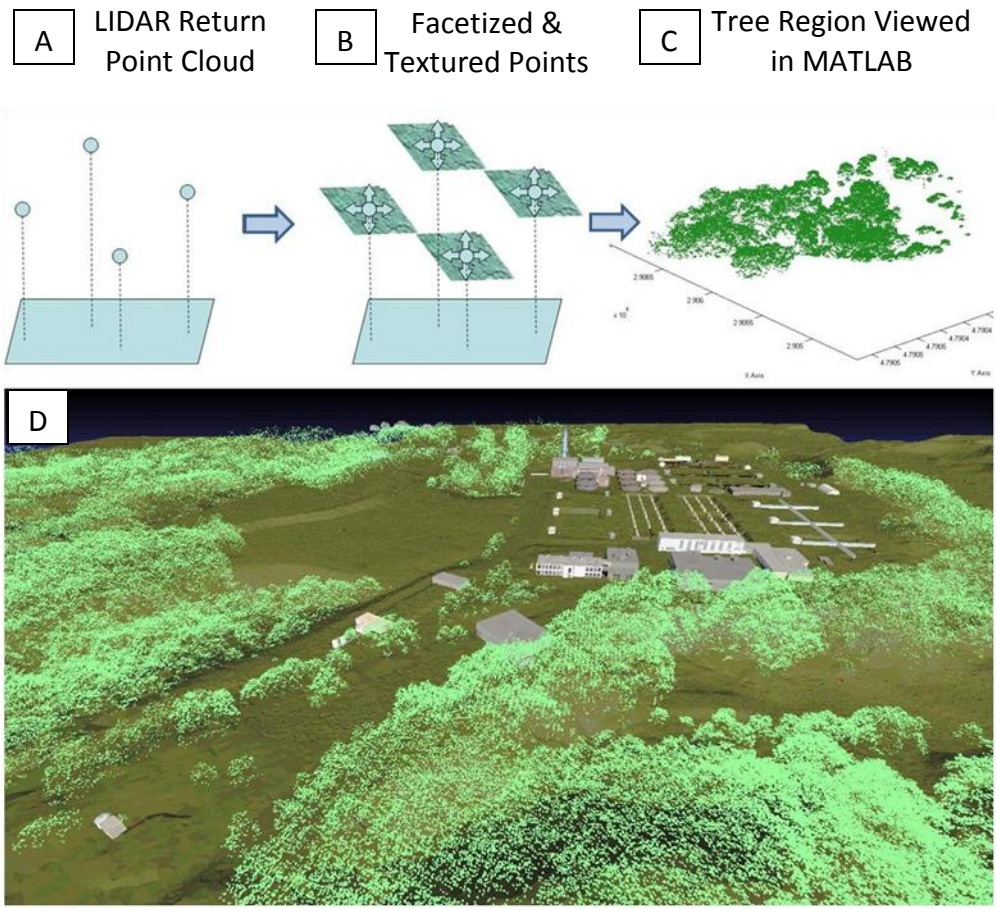


Figure 6-12 The process by which a LIDAR Return Point Cloud (A), can be transformed into model facets textured with real imagery of the forested terrain (B). The results of this process can be viewed above in MATLAB (C) or Meshlab (D).

An example of how LIDAR point returns can been converted into tree facets using this process is shown in Figure 6-12; while the results, when viewed with the building and terrain data in Blender (Blender Foundation 2010), are shown below in Figure 6-13.

Figure 6-13 The final model of the VanLare site, as viewed in Blender, using manually derived multiview imagery building models (courtesy Pictometry Int.) and LIDAR derived terrain and tree models.

6.2.3 LIDAR Direct – Developing LIDAR models in DIRSIG

In the previous section, a faceted LIDAR point cloud (Figure 6-10b) was utilized to relate an existing image derived model to the collected data. In many cases, a model will not exist and an analyst will be forced to utilize only the data on hand for model creation.

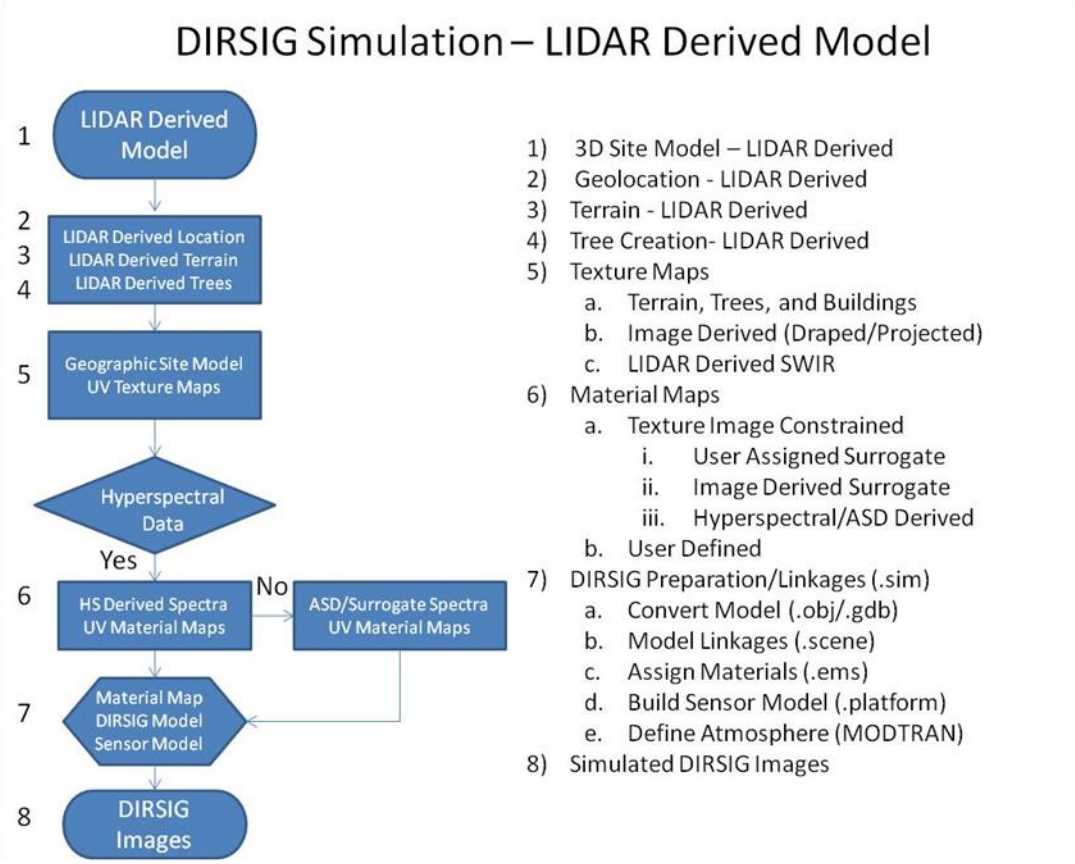


Figure 6-14 This flowchart depicts the process utilized for DIRSIG model creation using LIDAR data and imagery.

With a decent quality LIDAR dataset (~1m posting), a good representation of the site is still possible if the point cloud can be robustly facetized into a model utilizing techniques such as Delaunay Triangulation (Delaunay 1934). The author created MATLAB routine (Appendix G, Chapter 17) can read in a LIDAR point clouds directly and convert them into exportable ALIAS Wavefront 'OBJ' files (Bourke 2010), for use by most commercial 3D software packages. This code can be utilized to bring a coarsely modeled site directly into DIRSIG and has been thus dubbed 'LIDAR Direct' by the author.

A) LIDAR Point Cloud B) LIDAR Faceted Model C) LIDAR Model w/SWIR

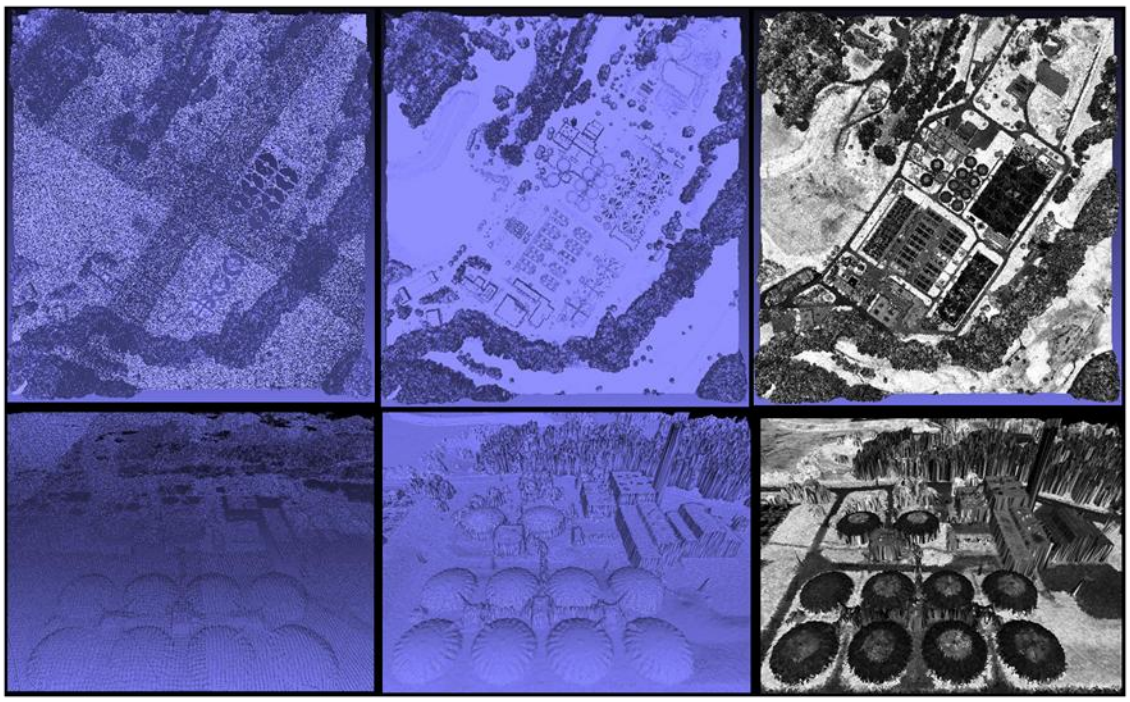


Figure 6-15 This graphics shows the 3 stages in transforming LIDAR data from a Point Cloud (A), to a faceted model (B), and finally texturing that model with the intensity return of the LIDAR itself (C).

6.2.3.1 *Draping Textures and Material Maps over the LIDAR Terrain*

Once the facetized model is generated, an associated texture and material map is still required for realism and material identification in DIRSIG. Since the same region of Rochester's MegaScene (Tile-4) was utilized for analysis, LIDAR data (from a collection over VanLare) was extracted for only that region. Thus, it was possible to directly associate the CITIPIX (Kodak Global Imaging 2008) imagery utilized as a texture and material map for that DIRSIG reference tile.

Now however, it is necessary to attribute the regions of the site that included buildings with the relevant, albeit potentially surrogate, material spectra. This process was relatively

straightforward due to the similar construction of many of the site buildings. At VanLare, the office buildings have crushed gravel roofs, the storage vats are covered with vinyl caps, and the pump buildings have white metal roofing. Thus it was a straightforward activity to generate regular shapes within a graphic arts package and "paint" them with an associated value for correlations within a LUT to the surrogate material of choice within DIRSIG. This process is highlighted below in Figure 6-16.

A) Site Imagery (Citipix) B) Terrain Segmentation C) Building Segmentation

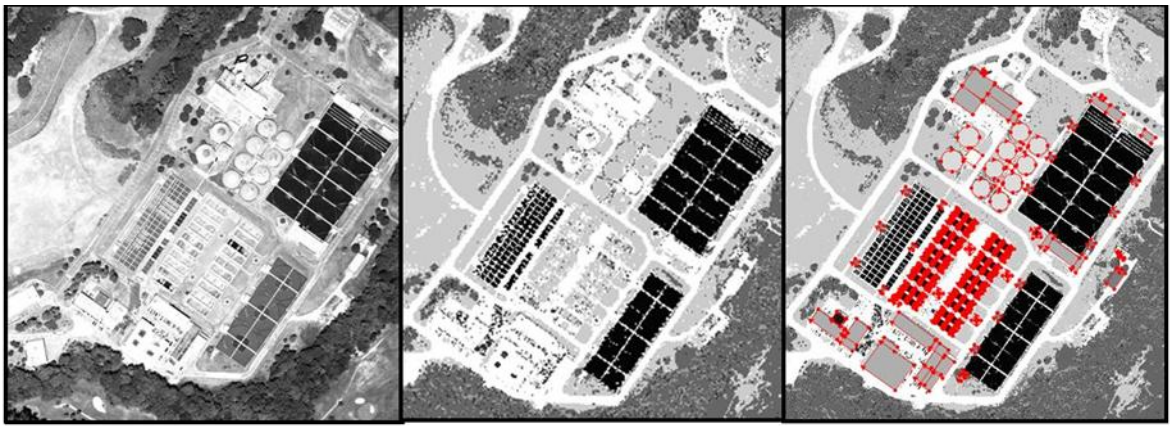


Figure 6-16 The LIDAR Direct process involves utilizing Imagery (A), to create a material map in order to physically describe the site. Here, automated segmentation of the terrain (B) is used in concert with user assisted ID of site materials (C).

A relevant point is that the material identification process is only necessary for activities that require DIRSIG simulations. By knowing the dominant materials in a scene it is possible to physically simulate representative atmospheric and illumination effects as well as various sensor collection modalities of interest. In Section 6.3.4, we will examine the DIRSIG simulations results using the LIDAR Direct approach to modeling a site of interest.

6.2.3.2 *Automatic Scene Object Identification Using LIDAR - Future Research*

Although not extensively tested by this author, it should be possible to automatically identify some of the scene's basic elements by utilizing the 3D spatial information of the LIDAR in concert with the SWIR return information. Segmentation of the scene into foliage (Kim, Hinckley and Briggs 2009), buildings (Gurram, et al. 2007), water, asphalt, and grass would allow a LIDAR developed site model to be directly ingested into DIRSIG with surrogate materials assigned to those structures.

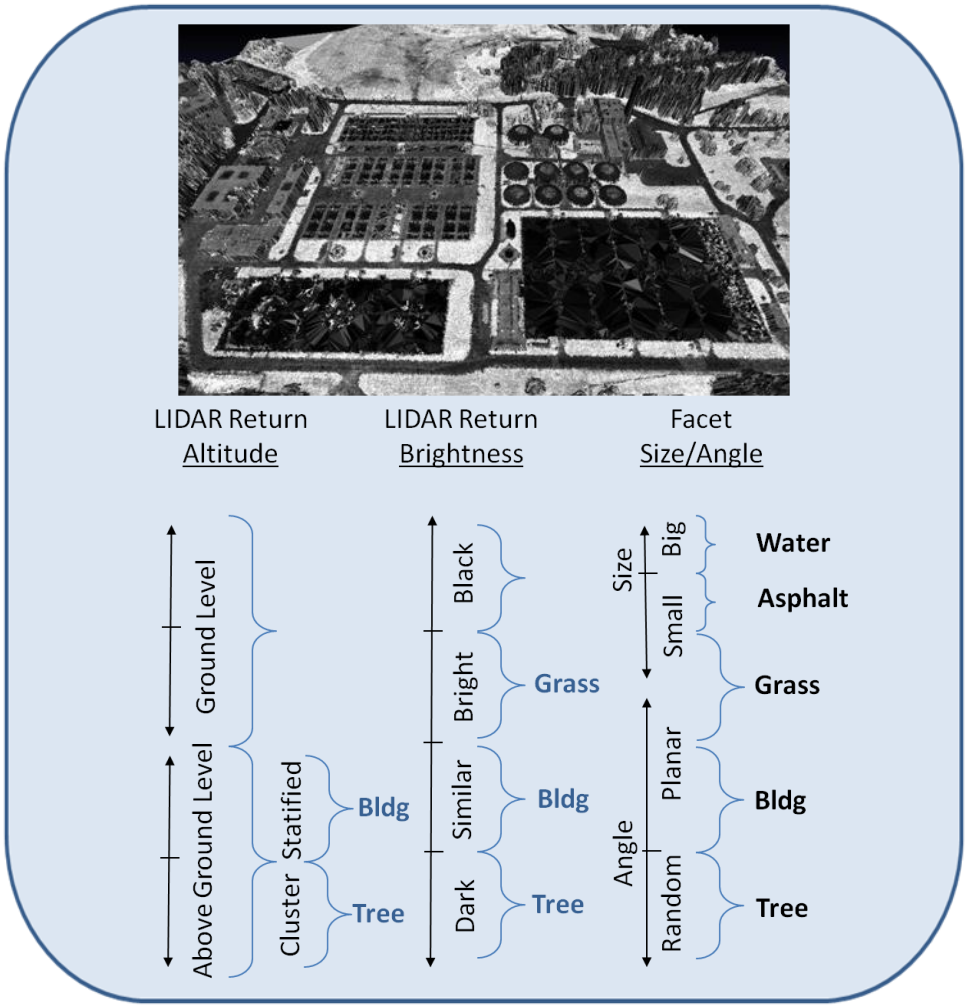


Figure 6-17 By using the spatial, brightness, and facetized characteristics of the LIDAR returns, aggregate material identification for DIRSIG should be possible.

A general approach to accomplish this activity is provided above in Figure 6-17. Once the aggregate classification of site materials is accomplished it can be assigned to the designated DIRSIG emissivity file, or a specific curve in that file, after an analysis of the histograms for each material characteristic. For instance, a facet designated as grass could be linked to a specific curve in the grass emissivity file (~400 curves available). This could be done by analyzing and relating the normalized brightness of all assigned grass facets w.r.t. the normalized distribution of emissivity curves. One could then assign the closest emissivity curve to the closest bin of brightness values.

6.2.3.3 *Material ID using Hyperspectral Sensing*

Finally, if Hyper-Spectral (HS) data from the site is available, it is possible to associate the resulting spectra directly to LIDAR facets or scene objects for material identification and physical modeling. This process would be very similar to the last section, where an individual HS data pixel curve could be associated to a specific LIDAR facet. Additionally, average spectra taken a region of interest (ROI), such as a rooftop, could be utilized as the material for a facet or grouping of facets. Of course, initial registration of the spectral data to the LIDAR data or geometric model would be necessary. By incorporating material identification from the HS data in concert with object identification from the LIDAR data (Figure 6-17) it would be possible to automatically perform some of these associations.

6.2.4 Imagery Direct - Developing Multiview Imagery models in DIRSIG

In Section 4.3.2, we examined techniques to recover 3D information solely from multiview imagery of a site. These techniques are essential for our 'model centric' approach to relating data, when an analyst only has access to 2D imagery. Since LIDAR data collections are still fairly

uncommon, due in no small part to the cost of current collection systems, the multiview imagery approach to geometric modeling may be the only avenue for deriving 3D characteristics of a site.

Although the techniques for depth recovery developed here can provide a relatively sparse reconstruction of a site compared to the dense reconstruction of LIDAR data, there is great promise in the ability of multiview imagery to provide high quality models using advanced reconstruction techniques (Pollefeys, et al. 2004). In order to accomplish this feat, a pixel-to-pixel mapping of the image overlap areas are required (Section 4.3.3.1). Although this is challenging to accomplish, due to the effects of occlusion and noise, a 'model-centric' archival of the bundled images may be necessary. In this scenario, an iterative model generation process could be utilized to self-rectify the images in order to help mitigate the effects of parallax and to help relate the images properly within a 3D construct.

An example of the relative quality of 3D site models that can be delivered via LIDAR DPCs, Multiview SPCs, and traditional Digital Elevation Maps (DEMs) is provided in Figure 6-18.

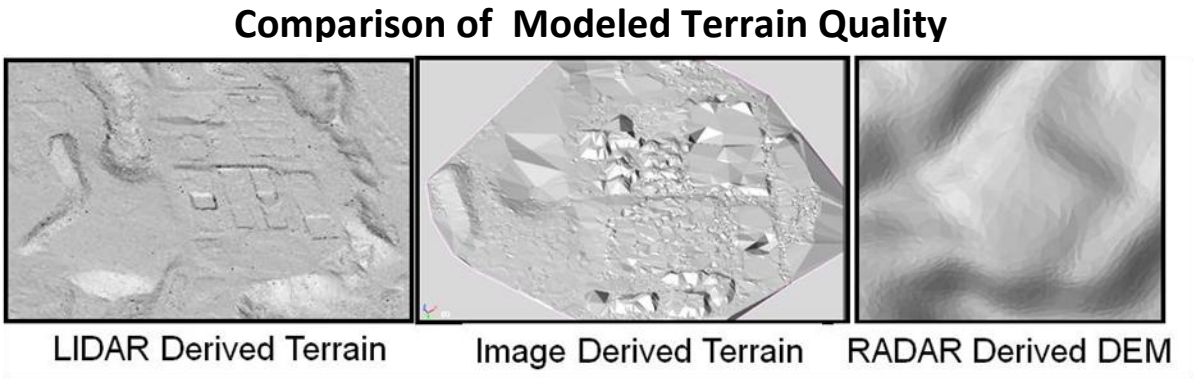


Figure 6-18 The relative quality of terrain information as derived from LIDAR, Multiview Imagery, and RADAR respectively.

Even the casual observer can see that Multiview Imagery can be utilized to provide terrain surface models that are much better than commonly available DEMs (~30m postings) and not much worse than LIDAR derived terrain, even with the 'first generation' sparse point clouds generated automatically using the techniques of Chapter 4. Additionally, even though the building structures may appear crude when compared to the Hybrid or LIDAR models of the previous sections, they are geospatially accurate enough to help place handmade models correctly. Finally, there is also great potential in the ability to use these Surface Elevation Maps (SEMs) to orthorectify imagery to a much higher accuracy than is now possible with DEMs. Although not covered here, the registration benefit of post rectified SEM imagery, in-order to mitigate 3D scene-to-sensor effects and estimate shadows, is an area the author recommends for high value future research.

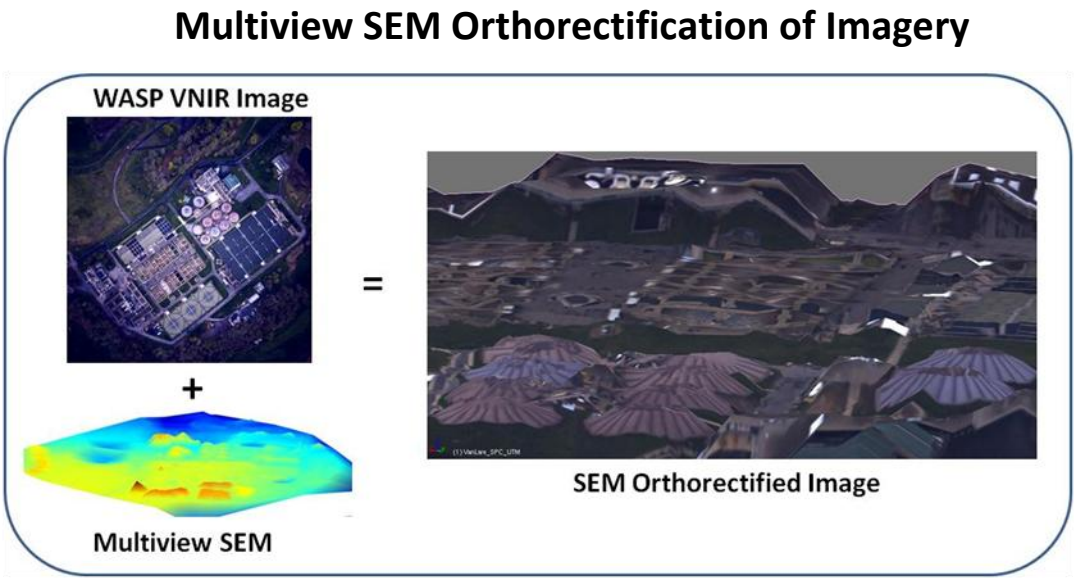


Figure 6-19 The ability to use Multiview Imagery derived Surface Elevation Maps to orthorectify an image is shown above.

6.3 Simulate – Physically (DIRSIG)

As mentioned earlier in this chapter, DIRSIG is the physics based simulator that will be utilized for capturing the 3D influences of the scene-to-sensor viewing geometry as well as estimating the multimodal appearance of the remotely sensed imagery. This software program has been developed over the last 20+ years by the dedicated staff of the Digital Imagery and Remote Sensing (DIRS) group, within the Center for Imaging Science, at the Rochester Institute of Technology.

Due to this groups steadfast research into understanding the physical underpinnings of Imaging Science and hard work by staff and students, DIRSIG's capabilities have steadily improved over the years and it is now considered a national asset by many in the field, including both the commercial and government sponsors. This is due in no small part to its unique ability to simulate physically accurate images from a variety of imagery sensors.

The following is an extract from the DIRSIG user's manual (Digital Imaging and Remote Sensing Laboratory 2006):

> *"The DIRSIG model is a complex synthetic image generation application which produces simulated imagery in the visible through thermal infrared regions. The model is designed to produce broad-band, multi-spectral and hyper-spectral imagery through the integration of a suite of first principles based radiation propagation models including the Air Force's MODerate resolution atmospheric TRANsmission (MODTRAN) program.*
>
> *First principles based approaches imply that fundamental physics, chemistry and mathematical theories are used to predict higher level phenomenologies. For example, the interaction between light and matter can be described using the work of Fresnel and others. These theories can be used to predict whether a photon with a certain wavelength will be absorbed or reflected by a material*

with a specific chemical composition. At a much higher level, the same interaction might be summarized as the "color" of the material.

Another example of a first principles approach would include the prediction of a surface temperature using fundamental properties including thermal conductivity, density, radiational absorption factors, radiational and convective loadings, etc. These parameters can be used with a set of fundamental governing equations that describe the flow of energy in and out of the surface to predict the steady-state temperature."

As mentioned earlier the DIRSIG model produces imagery using a predictive engine that is built around this collection of first principles based models and when properly implemented, can accurately predict physical imagery phenomena. This is the rationale for using DIRSIG as a multi-modal Rosetta Stone for image registration. A top-level flow chart of DIRSIG's simulation process is shown below (Figure 6-20). Additional information on DIRSIG, including the digital version of the user's manual is available at: http://dirsig.cis.rit.edu/ .

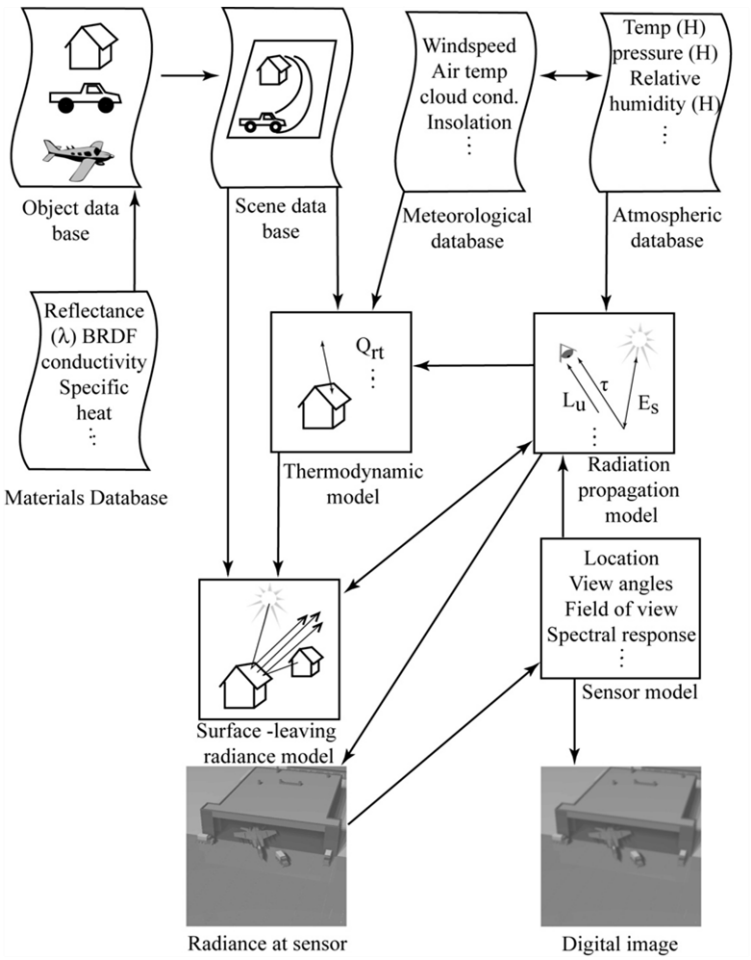


Figure 6-20 The physics based simulation process that DIRSIG utilizes for synthetic image generation (Digital Imaging and Remote Sensing Laboratory 2006).

6.3.1 Simulating WASP Imagery with DIRSIG

In order to compare multimodal imagery from the Wildfire Advanced Sensing Program (WASP) to simulated DIRSIG imagery, it is essential to model the environment, the imaging system, and the acquisition conditions properly. This is accomplished in DIRSIG through the use of a hierarchical file structure that that links different modules of the physical simulation to account for various elements of the Image Chain Approach (Schott 2007) analysis. A detailed example of how the DIRSIG files were arranged for the following simulations is captured in Appendix F (Chapter 16).

6.3.2 Simulating Materials and their associated Emissivity Curves in DIRSIG

An important consideration when simulating a scene within DIRSIG is in the application of material emissivity curves for proper physical simulation across multiple modalities. Often, it will be necessary to pull material emissivity curves from existing spectral libraries when field data from a spectrometer or hyperspectral data from an imaging sensor is not available. When a model facet/texture is known to be composed of a specific material, such as a gravel-covered rooftop, a surrogate spectra from an existing library can often be utilized to physically describe that object.

However, depending on the number of curves available in the emissivity file the associated scene texture may or may not be visually noticeable. This is because within a given material, the variability in texture digital count value (0-255) will be associated to specific curves in the emissivity file based on the Z-Score of the texture (Scanlan 2003). This concept is illustrated below in Figure 6-21.

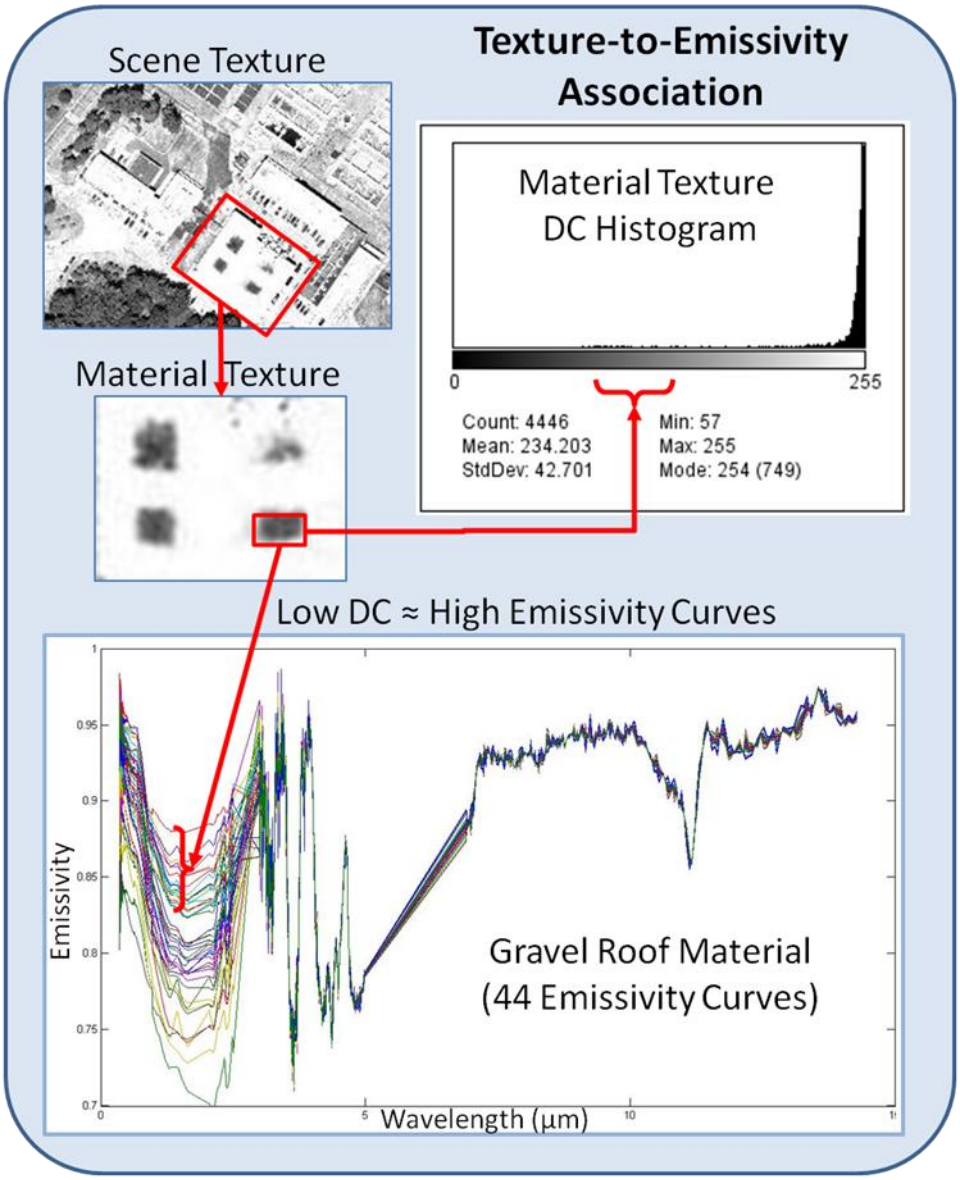


Figure 6-21 The general process involved when associating emissivity curves to intensity values from an image texture map. Here a region of interest was extract from the image and compared to the 44 curve emissivity plot (bottom) and the DC Histogram (right). Ideally, a simulation could link every DC value to a specific emissivity curve (i.e. 256 curves needed here).

So, if only one emissivity curve existed within an emissivity file to describe a material for use within DIRSIG, that material would appear to be a solid color with no texture variation appearing in that region of the scene. For the gravel rooftop example, a single "dark" emissivity curve would appear in the DIRSIG simulation as a solid dark gray color with no texture as seen below (Figure 6-22).

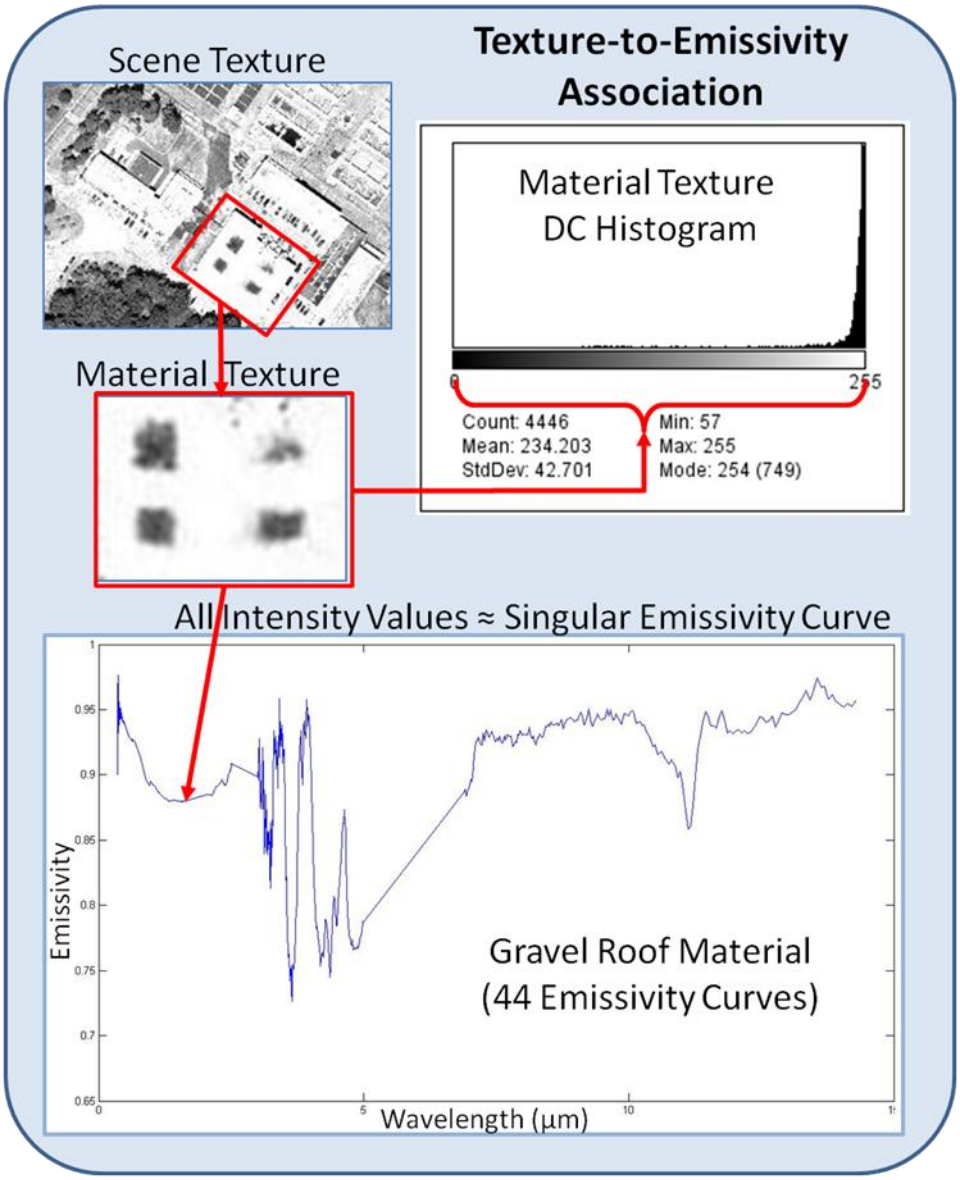


Figure 6-22 When only one emissivity curve exists in the material file, all of the image texture intensity values will be associated with only the singular curve. This will result in no texture information "coming through" in the DIRSIG simulation.

Since the number of curves utilized to represent a given material can have dramatic results in how well a DIRSIG simulation represent reality, it will often be necessary to take an existing material file and expand the number of emissivity curves. In essence, this takes the collected material spectra, taken under various viewing conditions, and increases the intensity diversity while maintaining the existing spectral character. In this way, it is possible to correlate an

individual DC intensity value to a specific emissivity curve, if at least 256 curves are generated. For additional information on the emissivity expansion utility provided with DIRSIG, please reference the associated documentation (Digital Imaging and Remote Sensing Laboratory 2006). A comparison of the results from an emissivity file (gray gravel) that was expanded from 44 curves to 400 curves is provided below in Figure 6-23.

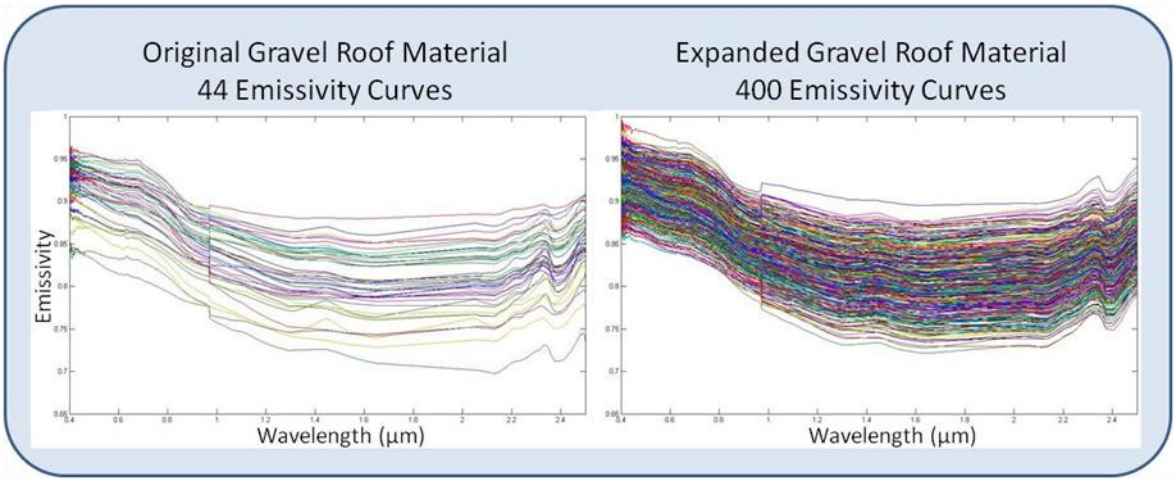


Figure 6-23 The resulting emissivity expansion of the original gravel roof material from 44 curves to 400.

Finally, a comparison of the DIRSIG results when running the same simulation, but using a singular emissivity curve versus one with 400 curves for the gravel roof material is shown in Figure 6-24.

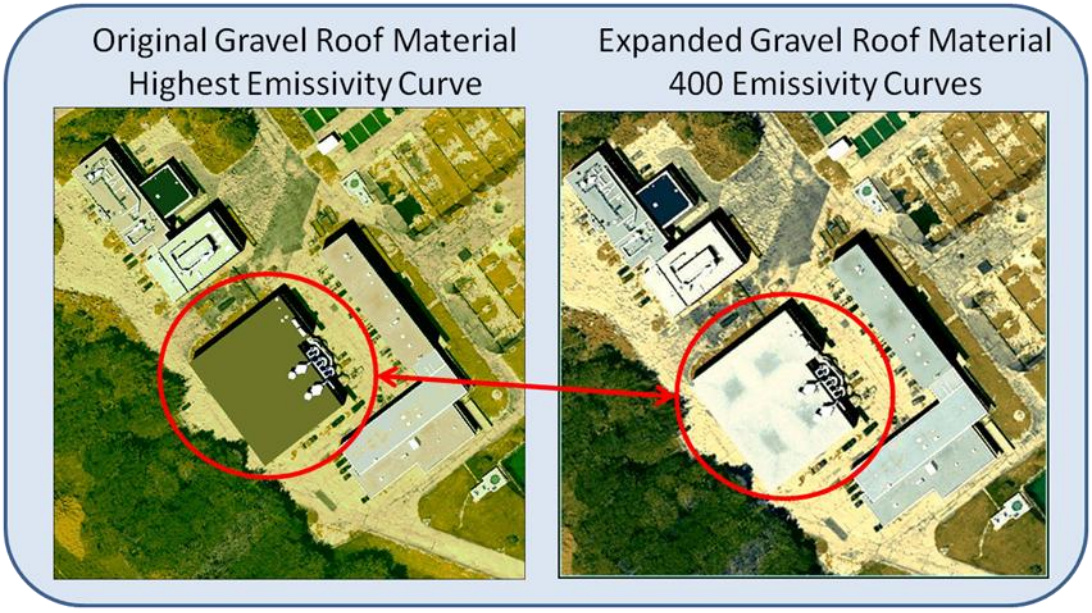


Figure 6-24 The simulated DIRSIG images above illustrate the need for material files with numerous emissivity curves to allow proper reconstruction of image texture within a scene.

6.3.3 Example DIRSIG Simulations of the Hybrid Model

The following figures show the resulting DIRSIG simulations from the hybrid modeling process. Recall that this process included a hi-fidelity image derived model (Pictometry 2010), LIDAR Derived terrain and trees, and airborne film based CITIPIX texture maps. First, in Figure 6-25, an oblique view of the hybrid model shows the detail on the sides of buildings at the VanLare Water Processing plant and although the tree creation process described earlier works well from near-NADIR view angles their horizontal facets reduce in size due to the cosine viewing effect.

Oblique View of the VanLare using the Hybrid DIRSIG model

Figure 6-25 The Hybrid DIRSIG model of the VanLare Water Processing Plant shown at an oblique view. From this vantage it is possible to see the detail on the sides of buildings, but, the tree facets are reduced in size due to the cosine viewing effect.

In Figure 6-26, the Southern and Northern sections are "zoomed in" for a closer look at the detail of the piping, building textures, and surrounding foliage.

Oblique View of the Southern and Northern sections of the VanLare plant.

Figure 6-26 In the figure above, the Southern (left) and Northern (right) sections of the VanLare plant are again visible at an oblique angle, but, now in slightly greater detail.

In the figure below (Figure 6-27), the near-NADIR DIRSIG simulation is meant to replicate the image collected by the WASP imaging system to mitigate the 3D influence of the terrain.

Near-NADIR View of the VanLare plant taken from WASP and simulated by DIRSIG.

Figure 6-27 On the left is a contrast enhanced image of the VanLare plant taken by the WASP imaging system, while on the right, is similarly enhanced DIRSIG simulation of the same site using the WASP view and the Hybrid model of the site.

By mimicking the same sensor-to-scene viewing geometries, it is possible to remove most of the 3D parallax effects that normally hinder automated image registration. The Northern region of the plant can be seen in greater detail in Figure 6-28 below.

The Northern VanLare plant imaged from WASP and simulated by DIRSIG.

Figure 6-28 The Northern portion of the VanLare Plant around the Smokestack and storage vats, imaged by WASP (left) and simulated by DIRSIG (right).

Similarly, the Southern section of the plant is shown in greater detail below in Figure 6-29.

The Southern VanLare plant imaged from WASP and simulated by DIRSIG.

Figure 6-29 The Southern portion of the VanLare Plant around the administration buildings, imaged by WASP (left) and simulated by DIRSIG (right).

Finally, an example of a SWIR image as taken by the WASP sensor and then compared to the simulated DIRSIG view in the appropriate spectral wavelength (Figure 6-30). Once the model is accurately generated, both geometrically and physically, it is a straightforward activity to change the sensor view or imaging characteristics to simulate the site from any angle across a diverse range of the imaging spectrum.

The VanLare Plant: Imaged from WASP in the SWIR and simulated by DIRSIG.

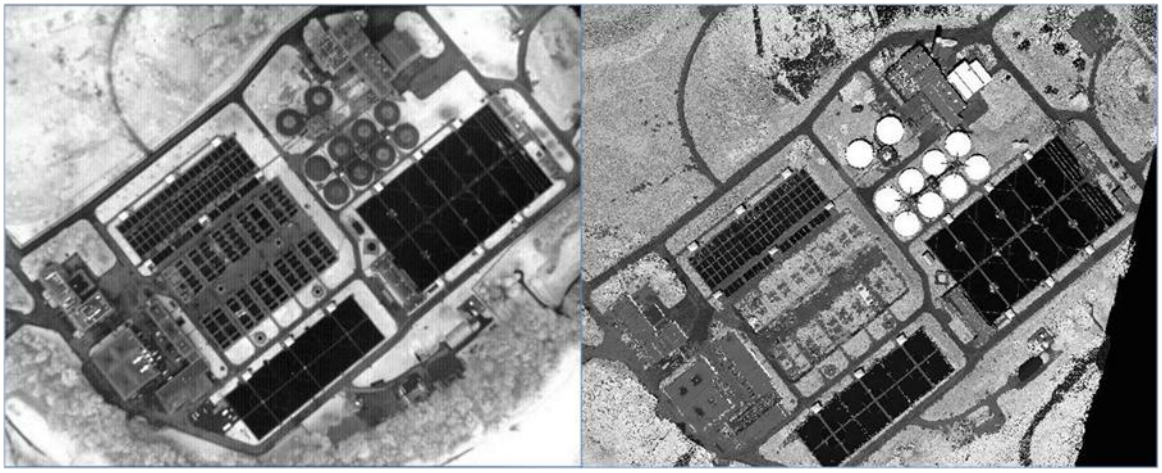


Figure 6-30 On the left is an image of the VanLare plant taken by the WASP SWIR sensor, while on the right, is a DIRSIG simulation of the site, in the same spectral region, using the WASP view and the Hybrid model of the site.

Note how the surrogate vinyl material used to represent the round storage tanks (upper center of each image) did not capture the inherent transition of the real material into the SWIR region of the spectrum. Since the standard vinyl material in the DIRSIG emissivity file database only has one curve, the author attempted to merge this data with an actual collection ASD spectrometer collection and then perform an emissivity expansion with only limited results. To capture the real physical essence of this material (for a better SWIR representation), several additional field collects would be required without blending in the "stock" database emissivity curve.

6.3.4 Example DIRSIG Simulations of the LIDAR Direct Model

The following image illustrates the resulting product of the LIDAR Direct approach to scene modeling and simulation using DIRSIG.

A) DIRSIG Terrain Map B) Site Material Association C) DIRSIG Simulation

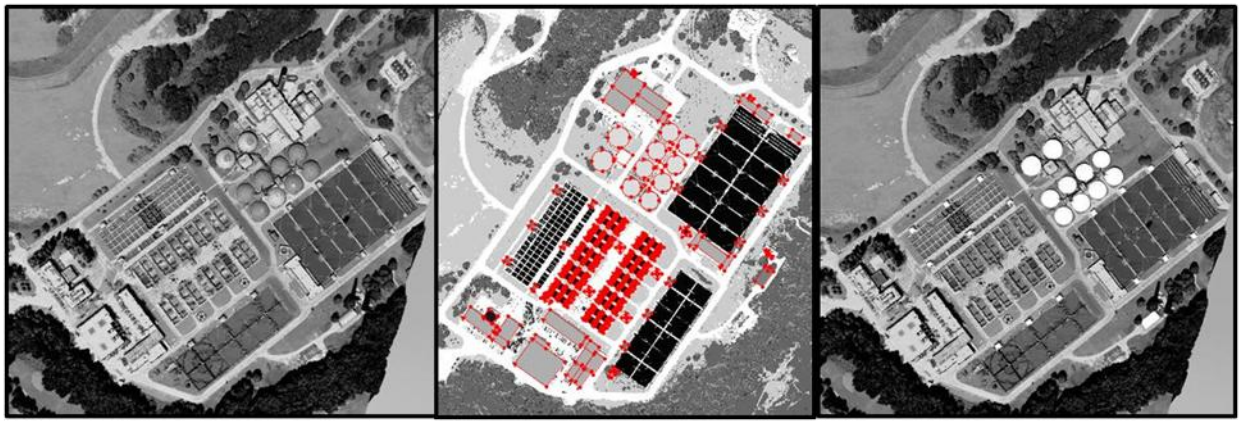


Figure 6-31 The LIDAR Direct process involves utilizing Imagery Textures and Materials Maps (A), with user assisted identification of dominant site materials (B) for ingestions into DIRSIG to physically simulate the site (C).

A grayscale version of a visible spectrum LIDAR Direct DIRSIG simulation compared to a grayscale WASP VNIR image is shown below in Figure 6-32. Since automated image registration still occurs predominantly in the grayscale regime, it is instructive to view the similarities between the simulated and real images as shown.

A) DIRSIG LIDAR Direct Simulation B) WASP VNIR Image of VanLare

Figure 6-32 The LIDAR Direct DIRSIG simulation's similarity to real imagery is readily apparent. The ability to relate LIDAR derived models, textured with archival imagery, to newly acquired images is key to the model centric approach.

When this same LIDAR Direct model is utilized in a DIRSIG simulation of the SWIR region, the similarity to real sensed data is evident when compared to an adjacent WASP SWIR image (Figure 6-33). Of additional interest is a visual comparison of the VNIR simulations (Figure 6-32) to the SWIR simulations (Figure 6-33), where the contrast reversal of the open water at the VanLare site is again evident.

A) DIRSIG SWIR Simulation B) WASP SWIR Image of VanLare

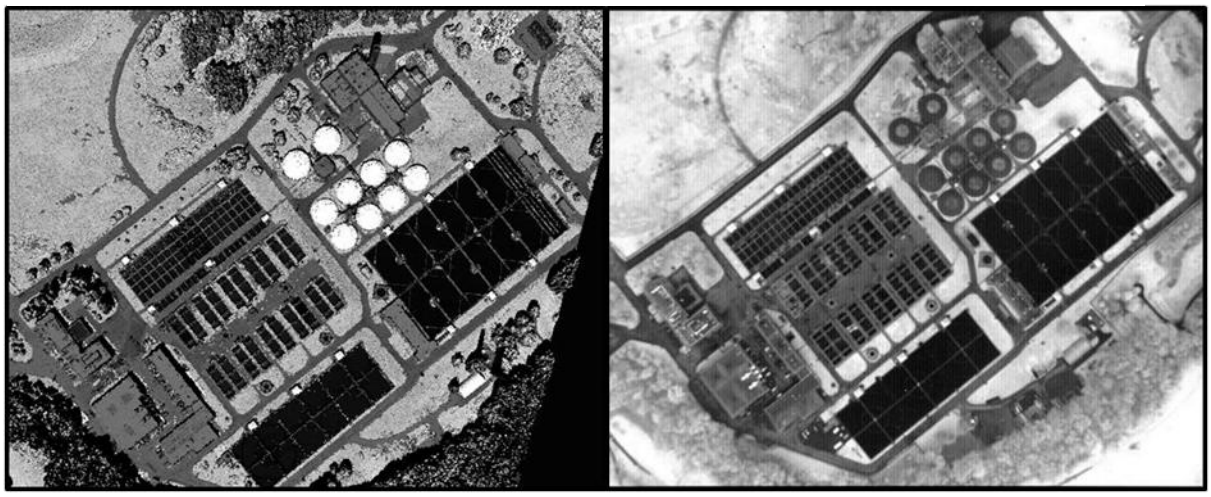


Figure 6-33 DIRSIG simulated image in the SWIR region (A) compared to an actual image from the WASP sensor acquired in the same SWIR region and from a similar camera position and orientation.

6.4 Relate - Mathematically

Once a model has been accurately generated, attributed, and physically simulated, it can be utilized as a "Rosetta Stone" to mathematically relate disparate multimodal datasets at arbitrary viewing geometries. It is hoped that the modeled scene is similar enough in both structural and spectral character to automatically relate via correspondence generation and matching techniques similar to the ones covered in Chapters 2 & 3. Since the DIRSIG modeled scene can be referenced to the world coordinate system, any datasets that are related in this manner can then be related to the global grid. The figure below (Figure 6-34) depicts how

image bundles created from “in-band” or “near-band” data, utilizing the techniques covered in Chapter 4, could then be related together utilizing this DIRSIG enabled technique, even if they are derived from disparate modalities.

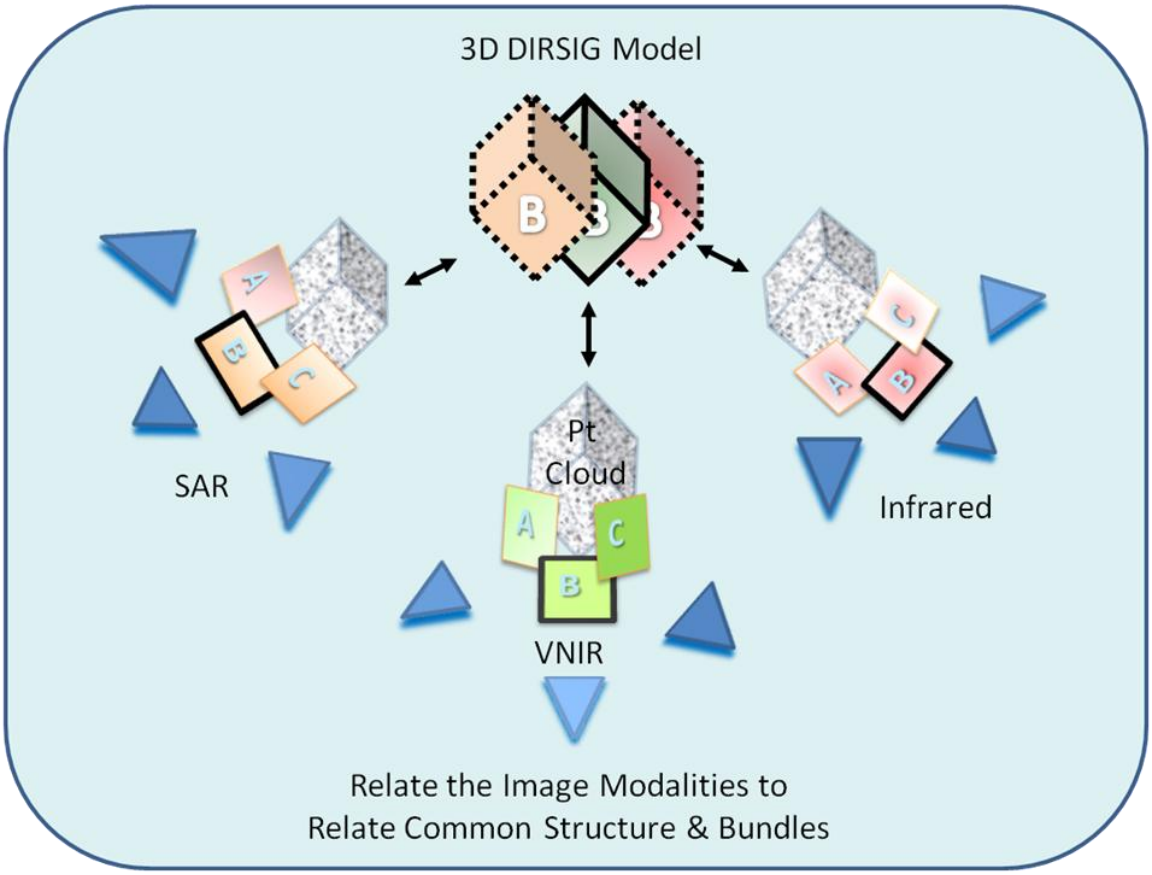


Figure 6-34 The basic process for relating multimodal image bundles utilizing DIRSIG. Here the model show various “colored” cubes that represent the 3D physical model which can be projected into an image of various modalities.

It is important to note that only a relatively small portion of the scene may require DIRSIG modeling, since only one image needs to be related to the synthetic scene to enable (potentially large) bundles of images to be related. Also, once accomplished, these images could form an “in-band” baseline texture for future registration, since the modeled scene can be used as the archive for registered images. This would allow for registration utilizing any

number of 3D modeling software or GIS systems (i.e. AANEE, MATLAB, Blender, GE, etc.), since they would only be utilized to account for the 3D sensor-to-scene viewing of effects and not the more challenging multi-modal appearance issues.

In the following example cases, the utility of physics based modeling to relate imagery datasets from the VNIR and SWIR modalities will be explored. First the hi-fidelity Hybrid DIRSIG model (Section 6.2.2) will be used to accomplish this feat and then the lower-fidelity LIDAR Direct DIRSIG model (Section 6.2.3) will be shown to have similar capabilities, albeit for the more restrictive NADIR imaging situations.

6.4.1 Error Analysis

As with most mathematical relationships, it is often of great interest to analyze the resulting model and understand how well the model fits the data. This is especially true in the area of image registration where error must often fall within a prescribed value for consideration as a "good result". In this field, subpixel registration accuracy is a common "gold standard" for results even though this criterion is often misunderstood and misapplied. The reason for this is that when a 2D mathematical model is used to relate images that are projections of 3D scenes, the best results can only approximate the true relationship. However, since this metric is in such prevalent use, it is necessary to understand how it is calculated and how it should be applied to the mathematical registration model of interest. Here we will be primarily concerned with conformal relationships, since most of the shear and projection effects have been removed through sensor modeling. Generally speaking, transformations should be

accomplished with the simplest mathematical relationship possible that is supported by the data.

6.4.1.1 *RMS Distance Error Calculation – Quantitative Error Analysis*

The Root Mean Square Distance Error (RMSDE) metric, as discussed briefly in Section 2.5.1, describes the deviation of selected correspondence feature locations from a 2D mathematical model, as measured in pixel distance. The matching correspondences are used to develop this 2D mathematical relationship and then each matched feature is tested for consistency with the model. In this metric (Eq. (102)), accuracy is judged based on the RMS Distance of the match location in the working image versus the location predicted by the model (which is derived from the total set of matched points). The average RMSDE of all the matches is used as the singular metric to define "goodness" of registration accuracy and is defined in Equation (103).

RMSDE Metric For 1 Matched Pair

$$RMSDE_{pair} = \sqrt{\frac{\left(x_{actual} - x_{predicted}\right)^2 + \left(y_{actual} - y_{predicted}\right)^2}{2}} \quad \textbf{(102)}$$

RMSDE Metric For the Total Image

$$RMSDE_{total} = \sum_{1}^{m} \frac{RMSDE_{pair}}{m} \quad \textbf{(103)}$$

Although 2D mathematical solutions are only approximations to the 3D registration problem, there is one important exception to this rule that will be exploited for our analysis. This exception is for situations when both images have been acquired from precisely the same position and orientation. In this case, the 3D projections of the scene onto the 2D focal plane

are the same and now the 3D problem can be solved accurately with a 2D solution. Although in many "real-world" situations this is an unattainable requirement, the author's model-centric approach to registration can take unique advantage of this principle. In other words, it is perfectly acceptable and probably advisable, to utilize a 2D mathematical model to relate the simulated image (of a projected model) to a real image since the 3D influences can been properly accounted for and recreated.

Normally the RMSDE metric can be used as an approximate measure of registration accuracy, but, it is only truly accurate for image registration in the following three situations:

A) Planar relationship exists in correspondence area (i.e. parking lots, floors, or sides of buildings), here the mathematical model may only provide a good localized relationship
B) Similar acquisition parameters due to small sensor movement or repeated views from a stable platform (i.e. video frames or satellite images from same location/orientation)
C) Simulated acquisition similarity using a modeled scene (i.e. DIRSIG Model-Centric approach)

6.4.1.2 *The Flicker Test – Qualitative Error Analysis*

The "Flicker Test", where the base image and registered working image are repeatedly overlaid visually can often give the user a better understanding of how well a registration has performed, but, this approach is often unquantifiable except through extensive human testing. However, when done properly (under proper Human Visual System (HVS) testing conditions), minute changes can be perceived. When using the *National Imagery Interpretability Rating Scale* (NIIRS), as little as 1/10 of a NIIRS can be a detected using the Flicker Test (Fiete and Tantalo 2001). Unfortunately, although these small visual errors in registration can be

perceived by most people, it is primarily qualitative and therefore of less value under automated situations except as a visual “quality control” measure.

6.4.2 Image Registration to the Hybrid DIRSIG Model

In Section 6.2.2 we developed a hi-fidelity DIRSIG model of the VanLare site in order to allow multimodal registration of imagery to that physical model from any vantage point. Below is an example using this DIRSIG model, which was textured using CITIPIX imagery in the visible region, to automatically register WASP SWIR imagery (Figure 6-35).

The registration and error analysis process begins with utilizing the SIFT algorithm to isolate scale invariant features within both images and correlating them as initial matches. In the figures below, SIFT identified 16 possible matches (Figure 6-35a). It is important to remember that this initial match list only represents similarity in image gradient features, not mathematical model consistency.

Next we utilize RANSAC in conjunction the Fundamental Matrix to ensure that epipolar constraints are maintained between the image pair; this results in the culling of 4 initial matches and we are left with 12 matches (Figure 6-35b). At this point the registration results already show a subpixel relationship, with the RMSDE = 0.87 [pix] as shown in Table 5.

Occasionally, it is possible to get an errant match that just happens to fall along its related epipolar line. For this reason, the author often finalizes the outlier removal process by filtering with RANSAC in conjunctions with the 2D Homography (Figure 6-35c). This can be robustly implemented in this situation since the 3D scene influences have been removed through simulation.

Although subpixel registration accuracy was accomplished after these outlier removal steps, it may be desirable to remove a few extra matches to improve the accuracy even further. The refinement process used to accomplish this task is handled via RMSDE error analysis (Walli, Multisensor Image Registration utilizing the LoG Filter and FWT 2003). This process is accomplished by iteratively culling the match that has the largest RMSDE and then re-computing the new model and related match errors. This process terminates when a desired total RMSDE is achieved for the remaining matches or when there are too few matches to compute the chosen mathematical model. In Figure 6-35d, only one additional match was culled to improve the accuracy of the model (Table 6) and to a limited degree, the shape of the cumulative RMSDE distribution curve (Figure 6-36). The resulting Homography (Conformal 2D Transformation) can be viewed below.

$$H = \begin{bmatrix} 0.996 & 0.0018 & 8.2272 \\ -0.0018 & 0.996 & -33.8784 \\ 0.0 & 0.0 & 1 \end{bmatrix}$$

Although the trained eye can see that this result demonstrates little influences of rotation and scale, it is possible to compute these values precisely by using Eqns. (27)-(30) as seen below:

$$Rotation\ (\emptyset) = \ 0.0018\ [rad]$$

$$Scale\ (S_{ave}) = \ 0.996\ [pix]$$

$$Translation\ (T_x, T_y) = (8.321 \quad -34.0008)\ \text{[pix]}$$

These results are compelling, because they imply that only the effects of translation need to be removed to adequately relate the two images. This is important because it provides evidence

that the 3D influences have been accurately mitigated through the 3D modeling approach to registration, by properly modeling the sensor's viewing pose. Also, of great practical importance, is the fact that a simple 2D translation can now be performed to properly archive the image as a projected texture onto the 3D scene. This is important, because in 3D site modeling and archival scenarios (such as the AANEE model, Section 1.3) the projected image corrections can be easily incorporated as a simple Latitude and Longitude shift, instead of a full 3D model pose correction (Section 3.2.3).

Finally, these results can be utilized to infer that the number of matching correspondences is sufficient for the transformation that is required. With traditional image registration tasks, where the 3D influences have not been mitigated, dozens of matches would normally be desired to increase the chances of resolving the most common 2D planar relationship. This is not required here since we have clearly addressed the 3D effects and are now only concerned with (at most) a 2D Conformal Transformation Homography. This requires a solution for only 5 parameters ($Rotation$, $Scale_x$, $Scale_y$, $Translation_x$, and $Translation_y$), which can be obtain with only three good match correspondences, since we know the $[x, y]$ locations from each control point. In fact, our results show that only the translation parameters are of great consequence and so only one good correspondence is necessary to correct the registered image for final archival. This means that the dozen good correspondences that were automatically recovered are more than sufficient to solve for this required correction.

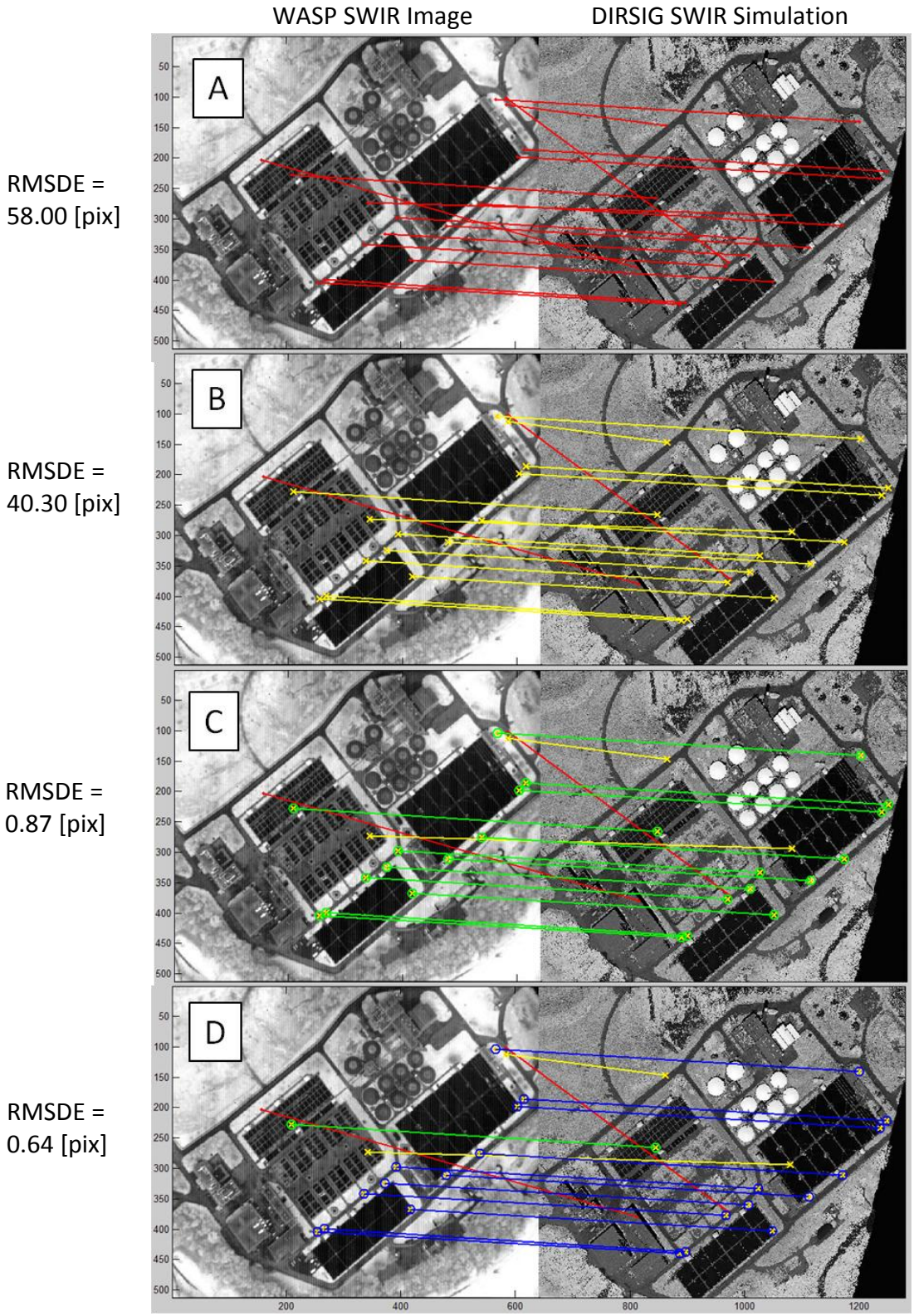


Figure 6-35 The images above show the initial WASP SWIR image paired with its DIRSIG simulation and the initial features matched using SIFT (A), the outliers removed using RANSAC with the F-Matrix (B), which were supported by using RANSAC with the M-Matrix (C), and finally where the largest contributing error match was removed using RMSDE analysis.

Sub-Pixel Accuracy achieved after removal of Match Outliers

Match #	X-Base	Y-Base	X-Work	Y-Work	X-Pred	Y-Pred	X-Error	Y-Error	RMSDE
1	565.70	104.29	560.45	141.10	559.21	140.25	-1.24	-0.85	1.06
2	254.59	403.65	246.40	439.66	247.38	439.76	0.98	0.10	0.69
3	391.71	297.45	384.12	333.45	384.79	333.53	0.67	0.08	0.48
4	266.00	399.29	257.07	437.29	258.81	435.40	1.74	-1.89	1.81
5	335.02	341.04	327.12	376.66	327.98	377.13	0.86	0.47	0.69
6	371.85	323.97	366.32	359.45	364.88	360.07	-1.44	0.62	1.11
7	538.25	275.90	531.55	311.60	531.57	312.08	0.02	0.48	0.34
8	209.28	228.13	204.59	265.58	202.16	263.94	-2.43	-1.64	2.07
9	614.94	186.07	608.12	221.94	608.45	222.19	0.33	0.25	0.29
10	416.04	367.33	408.42	402.84	409.09	403.53	0.67	0.69	0.68
11	603.39	198.47	597.04	233.89	596.87	234.60	-0.17	0.71	0.52
12	478.95	310.96	472.14	346.17	472.15	347.14	0.01	0.97	0.69
Mathematical Model = 2D Conformal						Total RMSDE [pix] =			0.87

Table 5 - The table above provides a breakdown of how the RMSDE Metric is computed for the previous example. Here the largest RMSDE contributor can be easily isolated and is highlighted in yellow.

Refined Accuracy after removal of Largest Error Contributor

Match #	X-Base	Y-Base	X-Work	Y-Work	X-Pred	Y-Pred	X-Error	Y-Error	RMSDE
1	565.70	104.29	560.45	141.10	559.48	139.72	-0.97	-1.38	1.19
2	254.59	403.65	246.40	439.66	246.58	439.74	0.18	0.08	0.14
3	391.71	297.45	384.12	333.45	384.44	333.35	0.32	-0.10	0.24
4	266.00	399.29	257.07	437.29	258.04	435.38	0.97	-1.91	1.51
5	335.02	341.04	327.12	376.66	327.45	377.02	0.33	0.36	0.34
6	371.85	323.97	366.32	359.45	364.46	359.95	-1.86	0.50	1.36
7	538.25	275.90	531.55	311.60	531.61	311.98	0.06	0.38	0.27
8	614.94	186.07	608.12	221.94	608.78	221.92	0.66	-0.02	0.46
9	416.04	367.33	408.42	402.84	408.75	403.56	0.33	0.72	0.56
10	603.39	198.47	597.04	233.89	597.16	234.35	0.12	0.46	0.34
11	478.95	310.96	472.14	346.17	472.01	347.07	-0.13	0.90	0.65
Mathematical Model = 2D Conformal						Total RMSDE [pix] =			0.64

Table 6 - By analyzing which matches contribute most to the RMSDE calculation it is often possible to iteratively cull the greatest error contributor; then recomputed the mathematical relationship and error to provide better registration results.

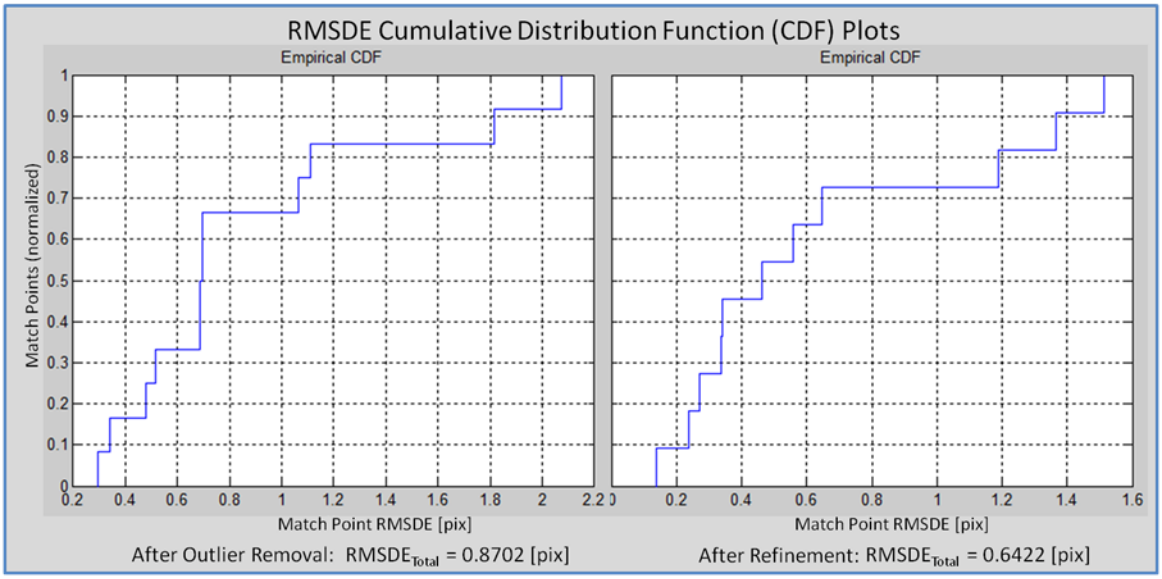


Figure 6-36 In the left plot, the initial RMSDE is plotted w.r.t. the number of good matches. After the largest error contributor was removed, the data was used to create a new model with error distributed slightly more linearly.

In this example, the 12 good matches (with total RMSDE below 1 [pix]) were automatically derived and then utilized to transform the DIRSIG simulated image into the same coordinate system as the WASP image. The results of this operation are visible below (Figure 6-37). In order to accurately archive these results, it would be necessary to reorient the DIRSIG model and then projectively texture the model with the WASP image (Section 6.4.4).

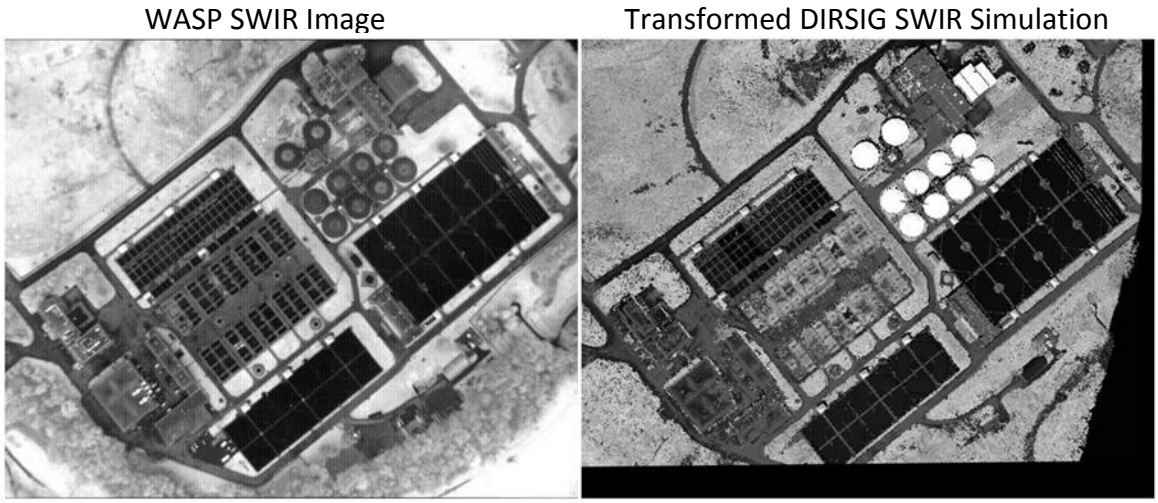


Figure 6-37 The results of the transformed DIRSIG simulated image (right), when compared to the WASP SWIR image (left).

This example demonstrates the power of physics based modeling, using DIRSIG, to account for the multimodal appearance differences between visible and infrared images of the same scene. Specifically, this approach was able to account for the multimodal, pose, temporal, and platform differences in the WASP and the CITIPIX imagery (used within DIRSIG for scene texture).

6.4.3 Image Registration to the LIDAR Direct DIRSIG Model

Similar to the process used above, we will now explore the utility of using the LIDAR Direct DIRSIG Model of the VanLare site (Section 6.2.3) to relate real multimodal imagery. As previously mentioned, this method allows an efficient modeling and attribution process within DIRSIG (hours vs. weeks) for users that have access to LIDAR or DPC multi-view data of a site of interest. The tradeoff for the inherent ease of modeling is in its more restrictive application to near-NADIR imaging scenarios. This is due to the lack of detail (texture and material attribution) on the sides of building models. However, with accurate sensor IMU/GPS knowledge this limitation could be addressed via projective texturing of a base image set onto the LIDAR data from various vantage points (i.e. using the Pictometry collection CONOPS).

WASP SWIR Image LIDAR Direct DIRSIG SWIR Simulation

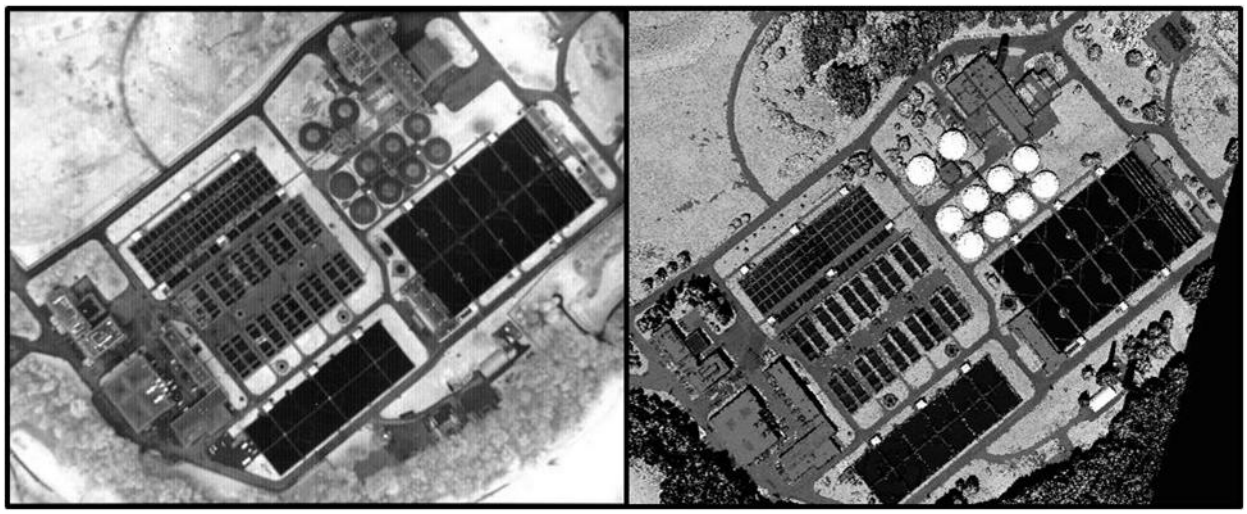


Figure 6-38 Here a WASP SWIR image of VanLare can be compared to the LIDAR Direct DIRSIG Simulation of the site.

An example of the LIDAR Direct DIRSIG model of the VanLare site compared to a WASP SWIR image of the same is shown in Figure 6-38. The utility for near-NADIR multimodal registration using the LIDAR Direct DIRSIG model as a multimodal "Rosetta Stone" is shown below (Figure 6-39). Here the registration process is visualized in steps that exemplify the process of extracting invariant features, relating these features, removing match outliers, and finally transforming the working image using the derived mathematical model from the good matches.

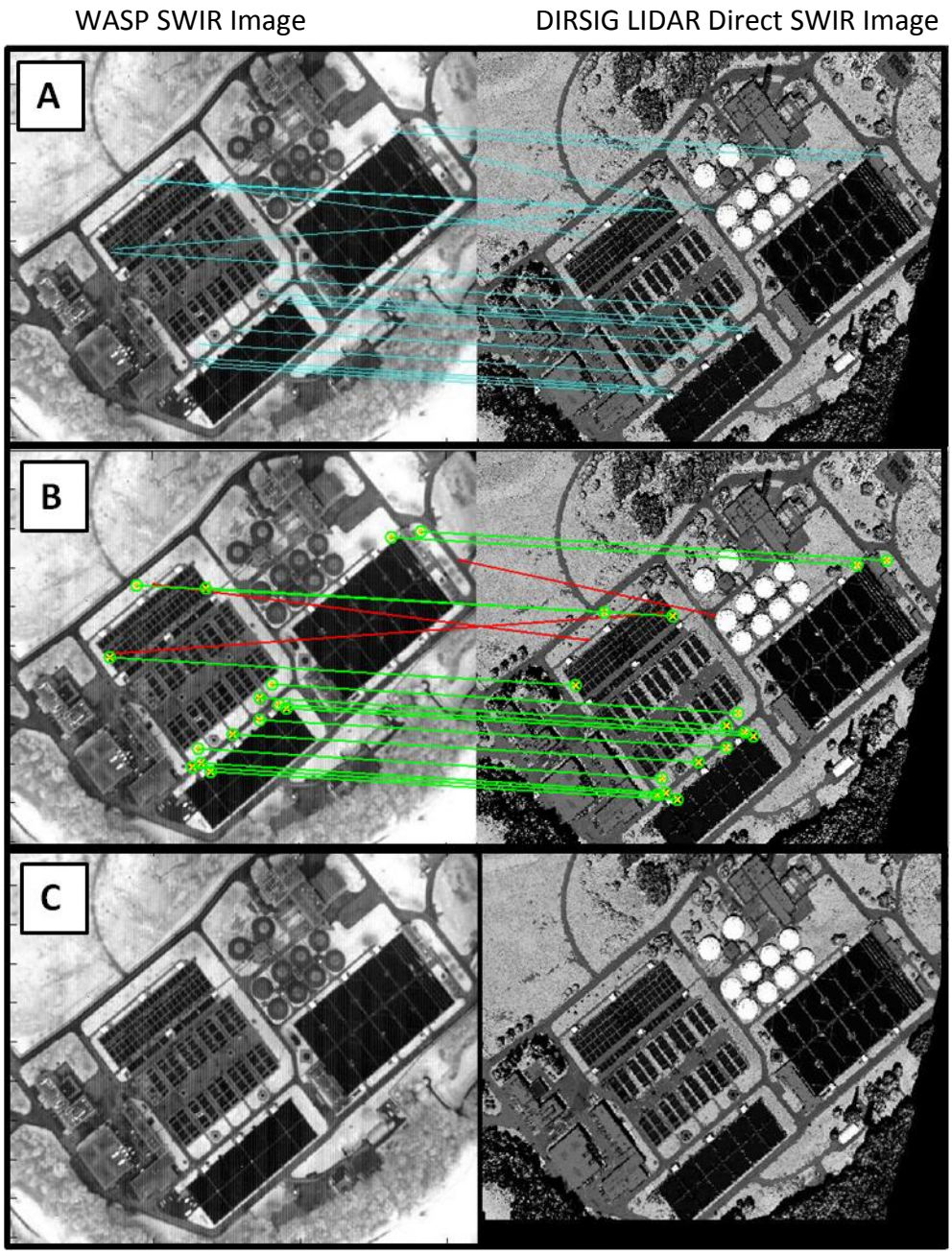


Figure 6-39 The Sequence above illustrates the features extracted using SIFT (A), outlier removal using RANSAC (B), and the final transformation using the resulting good matches (C), which resulted in sub-pixel registration accuracy.

The spreadsheet below (Table 7), illustrates how the RMSDE Metric is computed for a SWIR WASP Image which was registered to the LIDAR Direct DIRSIG Simulation of the same scene. These results were from the registration viewed in Figure 6-39.

Match #	X-Base	Y-Base	X-Work	Y-Work	X-Pred	Y-Pred	X-Error	Y-Error	RMSDE
1	262.13	381.27	253.35	418.3	255.6502	416.8895	2.3002	-1.4105	1.9079
2	363.47	299.24	356.12	335.14	357.3235	335.0507	1.2035	-0.0893	0.8534
3	565.7	104.29	559.52	140.62	560.3047	140.4491	0.7847	-0.1709	0.5678
4	254.59	403.65	246.46	439.46	248.0406	439.2725	1.5806	-0.1875	1.1255
5	383.16	329.46	376.98	364.81	376.9514	365.357	-0.0286	0.547	0.3873
6	179.12	172.9	174.81	207.11	173.1246	208.0695	-1.6854	0.9595	1.3713
7	279.9	410.55	274.37	446.09	273.3586	446.2495	-1.0114	0.1595	0.724
8	266	399.29	258.75	436.88	259.4748	434.9393	0.7248	-1.9407	1.4649
9	309.75	362.54	302.9	397.9	303.3723	398.2704	0.4723	0.3704	0.4245
10	346.27	344.62	340.92	379.72	339.9805	380.4317	-0.9395	0.7117	0.8335
11	371.85	323.97	366.15	359.47	365.6445	359.8301	-0.5055	0.3601	0.4389
12	279.37	410.31	274.37	446.09	272.8287	446.0078	-1.5413	-0.0822	1.0914
13	346.07	316.45	340.22	351.97	339.8577	352.2312	-0.3623	0.2612	0.3158
14	273.32	176.38	268.16	211.65	267.4153	211.8121	-0.7447	0.1621	0.5389
15	524.63	111.85	518.87	147.37	519.1702	147.9043	0.3002	0.5343	0.4333
16	141.99	264.79	136.25	300.14	135.7025	299.9552	-0.5475	-0.1848	0.4086
Mathematical Model = 2D Conformal						Total RMSDE [pix] =			0.805438

Table 7 – This spreadsheet provides a breakdown of how the RMSDE Metric is computed for the LIDAR Direct example.

6.4.4 Reorient the Model to Incorporate the Registration Results

The resulting 2d Homographies from both DIRSIG simulations (Section 6.4.2 & 6.4.3) will now be used as exemplars to show how the resulting 2D transformation homography (H_{3x3}) can be utilized to change the pose of the model for proper image texture alignment and archival.

The homography resulting from the 12 good match points of the Hybrid DIRSIG model and the SWIR WASP image (Table 5) is shown below (note the slight difference to the Homography presented earlier due to the desire to utilize the unrefined 'good matches' in both cases):

$$H_{3x3} = \begin{bmatrix} 0.9985 & 0.0009 & \mathbf{7.1729} \\ -0.0009 & 0.9985 & \mathbf{-35.2426} \\ 0.0 & 0.0 & 1 \end{bmatrix}$$

$$T_{xy} = [7.2156 \quad -35.2874]$$

As mentioned earlier, this transform contains virtually no rotation and shear (upper left 2x2 sub-matrix) and very minor scale influences (the diagonal of the upper left sub-matrix). This is to be expected, since great effort was placed in accurately modeling these influences in the DIRSIG model. In fact, utilizing a simulation to remove these effects provides a great deal of power and flexibility in the automatic registration phase and justifies the modeling process.

The homography resulting from the LIDAR-Direct DIRSIG model and the SWIR WASP image (Table 7), is shown below with very similar results:

$$H_{3x3} = \begin{bmatrix} 0.9989 & 0.0027 & \mathbf{5.61} \\ -0.0027 & 0.9989 & \mathbf{-34.4718} \\ 0 & 0 & 1 \end{bmatrix}$$

$$T_{xy} = [5.3311 \quad -33.4107]$$

Again, the only transformation required is a shift of the image simulation (by applying the inverse homography). This is then converted into meters and reinserted into DIRSIG or the archival software of choice for proper texturing of the image onto the model. Although the 3D models were constructed using very different techniques, the resulting transformation to relate the WASP image is quite similar. The overall RMS Error between the two translation results is:

$$Translation\ Error_{RMS} = \sqrt{((7.2156 - 5.3311)^2 + (-35.2874 + 33.4107)^2)/2} = 1.8806\ [pix]$$

6.5 Archive – Texturally (Map the Real Image to the Model)

The final phase in the MSRA process is archiving the acquired image to the site model. This is accomplished by recovering the new 3D pose of the model with respect to the image and projectively texturing the image onto the model.

6.5.1 Model Pose from Matched Features

It is possible to recover the model pose and position with respect to an image using the techniques covered in Sections 3.1 & 3.2. However, in this situation we have image-to-image matches as our input, not image-to-model correspondences. Fortunately, we have the associated 3D model that was used to create the simulated image from which the correspondences were derived. This can allow correlation of the closest 3D point once a ray is cast from the correspondence to the camera center of the simulated image. This concept can be visualized below in Figure 6-40.

Additionally, it is possible to implement mathematical techniques that link the camera orientation parameters directly to the 2D Projective Homography (Seedahmed 2006), as discussed in Section 3.1.2. This technique is especially applicable in this situation due to the legitimacy of the 2D RMSDE assurance of a good planar model fit to the final solution space. Using this technique, the image correspondences to the 2D projection of the model can directly provide the relative camera position and pose. So by keeping the image as the origin, the final model position and pose is simply the inverse transform derived from this interim solution.

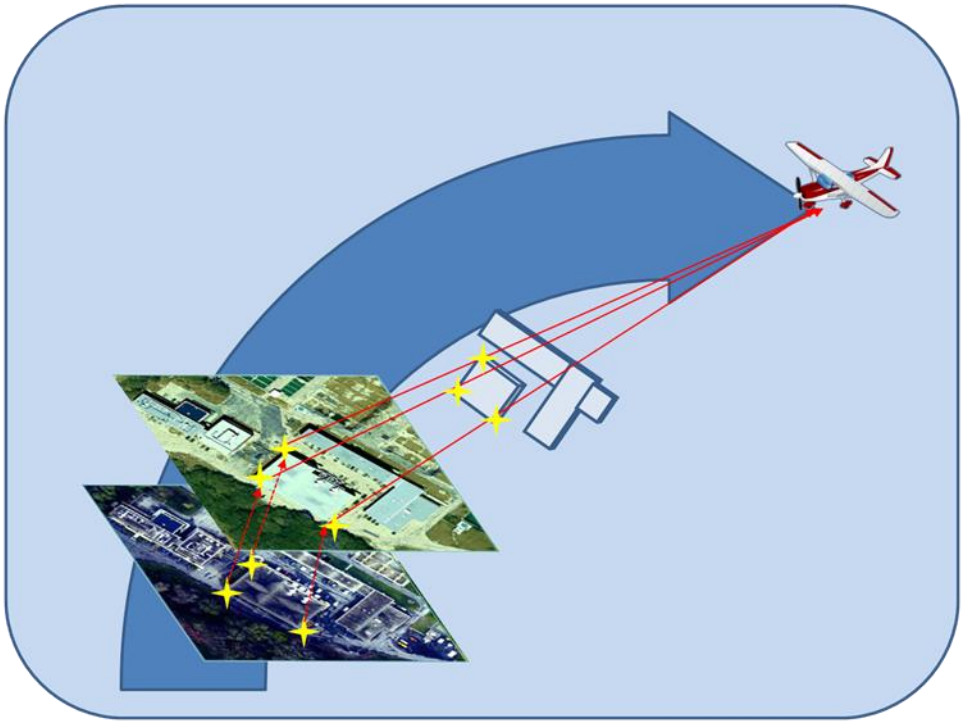


Figure 6-40 By ray tracing from the camera to the simulated image correspondence location it is possible to isolate the 3D model location of interest for use in pose estimation.

6.5.2 Projective Texture – Image to Model

In order to archive the newly related imagery to the 3D model, it is necessary to project the model onto the image and extract the model vertices as "vertex texture" locations for mapping into the uv plane. This is accomplished in a similar, but opposite manner to the previous section. Here a projection matrix is utilized to flatten a 3D model onto an image, to simulate a camera's view of the scene. Once this accomplished, the 2D projected model vertex locations can be utilized to directly associate the resulting image pixels as vertex texture locations. This process can be visualized below in Figure 6-41.

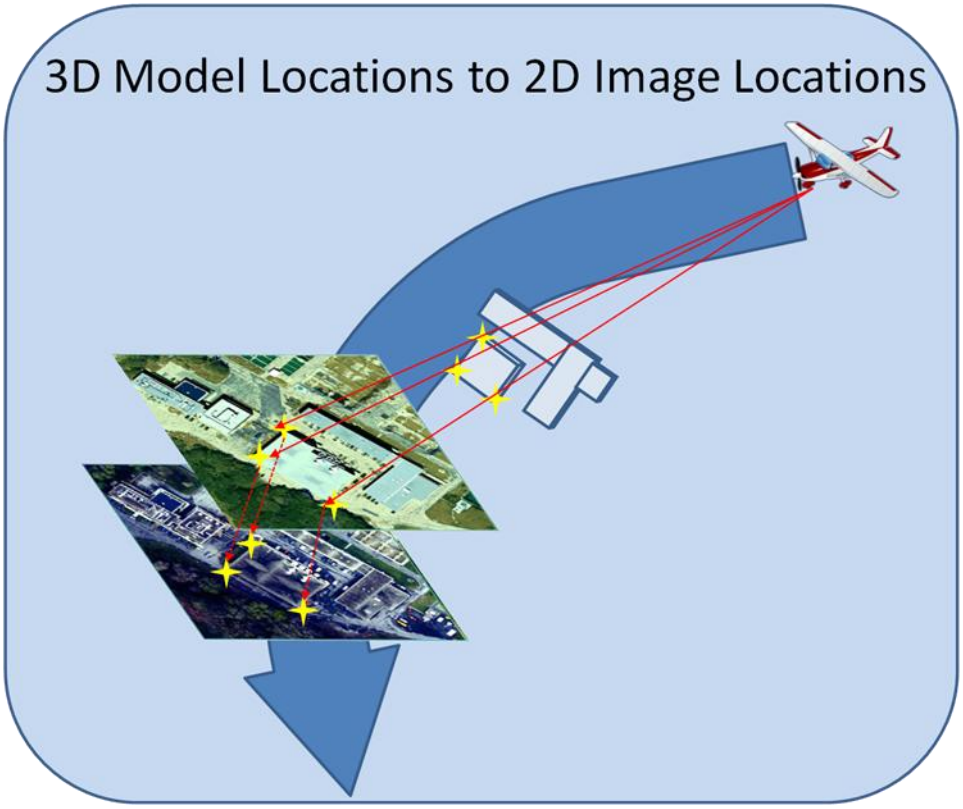


Figure 6-41 To obtain "vertex texture" locations for UV mapping a model to an image starts at the camera and then projects the 3D model onto a 2D image. The projected model vertex locations on the image are the uv texture locations.

6.5.2.1 *Creating UV Texture Maps for SPC Models*

The technique for relating SPC Models to images for creating UV Textures is relatively straightforward. Here the initial correspondences between the base image and the periphery images will be utilized to generate the Vertex Texture (VT) locations for the final UV Mapping. Since all the matches occur with the base image, it is only necessary to take these image locations and relate them to the final 3D model vertices that are derived from the image matches. This will take on the following format for the ".OBJ" model description:

Table 8 This table contains the related 3D model vertex and related image UV texture locations in the ".obj" format.

OBJ ID	Model X [m]	Model Y [m]	Model Z [m]
V	290610.83	4790142.40	108.86
V	290626.79	4790245.10	109.23
V	290629.49	4790247.50	108.99
V	290652.95	4790279.40	107.50
V	290645.96	4790306.80	105.19

OBJ ID	Image x [pix]	Image y [pix]
VT	1209.39	2755.81
VT	1560.34	2045.48
VT	1585.40	2035.36
VT	1827.59	1858.82
VT	1839.19	1650.06

In the ".OBJ" UV Texture format, the order is important; here the first vertex that is described with the letter "V" will be associated with the first vertex texture location described with the letter "VT". An extracted subset of five of the SPC model vertices (V), with their associated WASP image vertex texture locations (VT), is provided above in Table 8.

An example of how this works is shown in Figure 6-42, where the entire set of ~17 thousand correspondences, generated from the 5 images of the VanLare Processing Plant, were utilized in Chapter 4 to create a SPC Model of the terrain. These model and image location points were then related to generate a precise UV Texture Map for the model using the base WASP image. A closer look at these results is also available for reference in Section 4.3.3.3 (Figure 4-23).

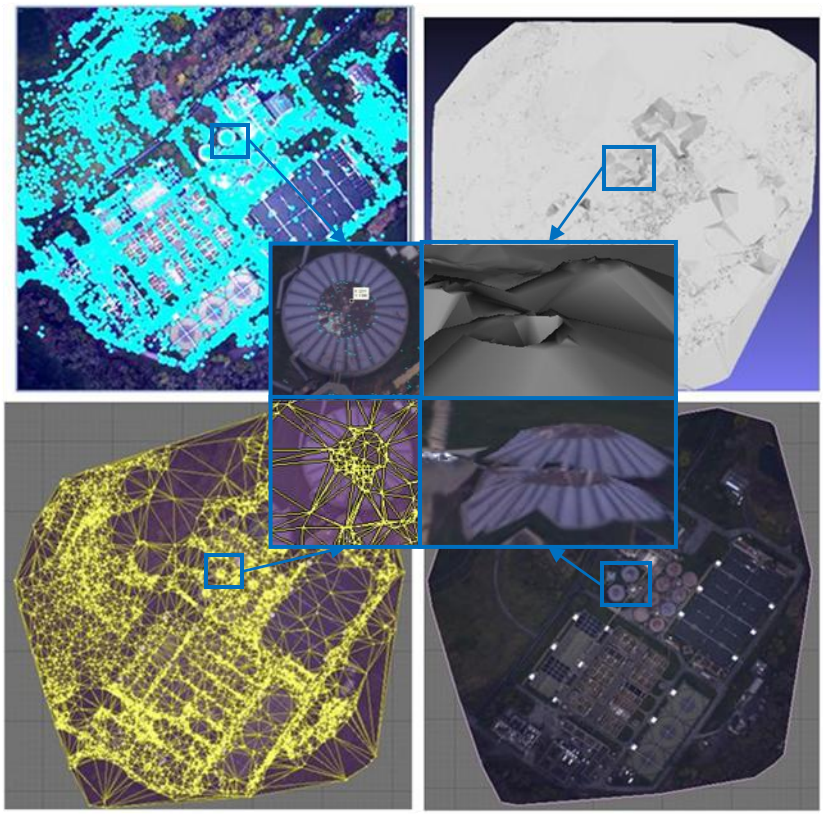


Figure 6-42 This series of snapshots show how the matches from the base image can be directly related to the 3D SPC model and then used as the vertex texture locations with the base image to create the model's UV Texture map.

6.5.2.2 *Creating UV Texture Maps for LIDAR and existing Models*

Unlike with the previous section, the scene model cannot normally be automatically associated with known image locations without having to first register the texture image with the projected model. However, if the model was generated from LIDAR data then a few options for automated process are available. First the model can utilize the inherent IR intensity return information to attribute the model. Once this is accomplished this attributed model can be automatically registered to SWIR imagery and then UV Texture Mapped.

In order to attribute a LIDAR derived scene model with the SWIR Intensity information from the pulsed return, the author has developed the following recipe which has been implemented in MATLAB:

Recipe: IR Attribution for LIDAR Models

1) Analyze the 3 Vertices from each model facet
2) Average the associate LIDAR SWIR DC Returns for all facet triplets
3) Rescale the ave. grayscale returns from integer values of 1 to 255
4) Assign the facet color to the replicated grayscale value [255 255 255]
5) Convert the vertices and vertex color information to ".OBJ" model format:

```
# Walli09 DIRSIG LIDAR Direct OBJ File:
        mtllib VanLare_composite_ROI_VNIR045.mtl
        v 291037.980000 4790028.000000 76.830000
        v 291037.950000 4790027.900000 76.850000
        v 291037.930000 4790027.900000 76.880000
        g 0
        usemtl 0
        f               1               2               3
        ...
        g 255
        usemtl 255
        f               300             301             302
```

6) Link the material ID's in the ".OBJ" with their descriptions in the ".MTL" file

```
# Walli09 DIRSIG LIDAR Direct MTL File:
# mtllib VanLare_composite_ROI_VNIR045.mtl
        Material Count: 256
        newmtl 0
        Ns 0.000000
        Ka 0.000000 0.000000 0.000000
        Kd 0 0 0
        Ks 0.050000 0.050000 0.050000
        Ni 1.000000d 1.000000
        illum 3
        ...
        newmtl 255
        Ns 0.000000
        Ka 0.000000 0.000000 0.000000
        Kd 1.0  1.0  1.0
        Ks 0.050000 0.050000 0.050000
        Ni 1.000000d 1.000000
        illum 3
```

This process will result in a 3D model that can be utilized within almost any modeling package due to the enduring popularity of the common ".OBJ" model standard. An example of a SWIR attributed LIDAR model can be seen below in Figure 6-43.

Direct Attribution of LIDAR Model facets using the IR Return

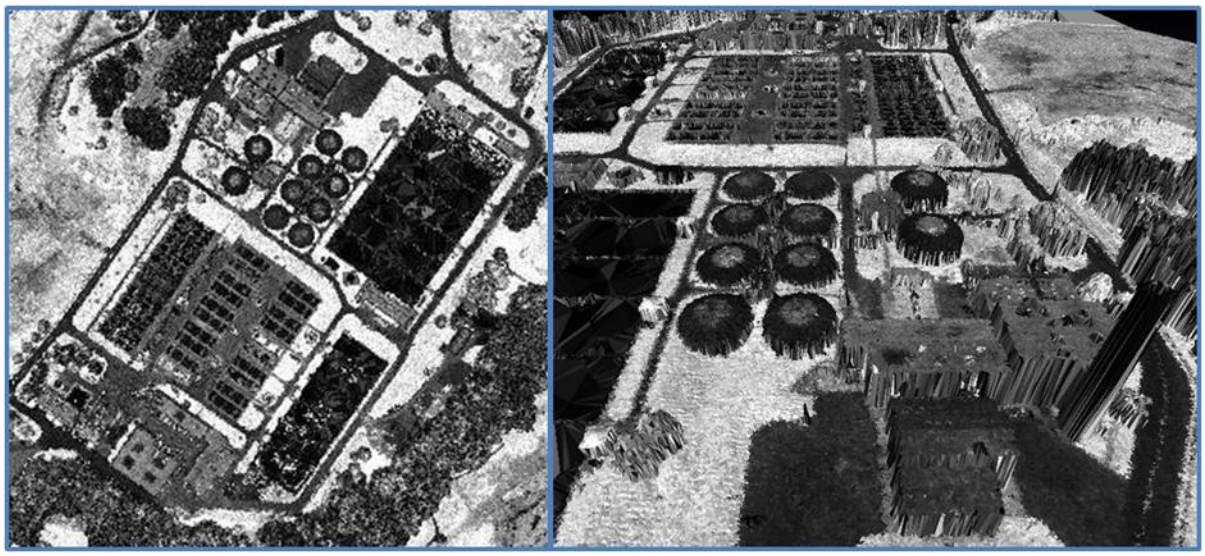


Figure 6-43 This figure shows the IR Attributed LIDAR model from a NADIR (right) and an oblique (left) view.

In Section 3.3, the ability to directly register a projected image of this model to an actual SWIR image (taken by the WASP sensor) was demonstrated. After the image has been registered to the model, the linear and nonlinear techniques represented in Section 3.2 can be utilized to reorient the model to align properly with the viewing geometry captured by the image. Once the accurate EOP have been recovered, they can be utilized with the Projection Matrix (P) to project the LIDAR model onto the SWIR image. Once this is accomplished the process for UV texturing is very similar to the one presented in the last section for assigning model vertices to the image locations.

6.6 Results Summary – DIRSIG as a Multimodal Rosetta Stone

In this chapter we have addressed the challenging area of multimodal 3D image registration through the use of physics based modeling. In order to provide nimble access to a variety of different modalities (VNIR, Infrared, SAR, Polarimetric, and LIDAR) the author has utilized the CIS Digital Imagery and Remote Sensing Image Generation (DIRSIG) software to physically model the VanLare site (Digital Imaging and Remote Sensing Laboratory 2006).

Automated multimodal registration of near-NADIR scenes has been demonstrated and oblique views should be possible when DIRSIG is used in concert with an accurate and properly oriented 3D scene model. The previous examples should provide sufficient evidence that using DIRSIG as a physical modeling based "Rosetta Stone" to relate multimodal imagery is not only feasible, but, advantageous due to its extensibility into various regions of the EMS. The following is a quick breakdown of these accomplishments into an explicit form.

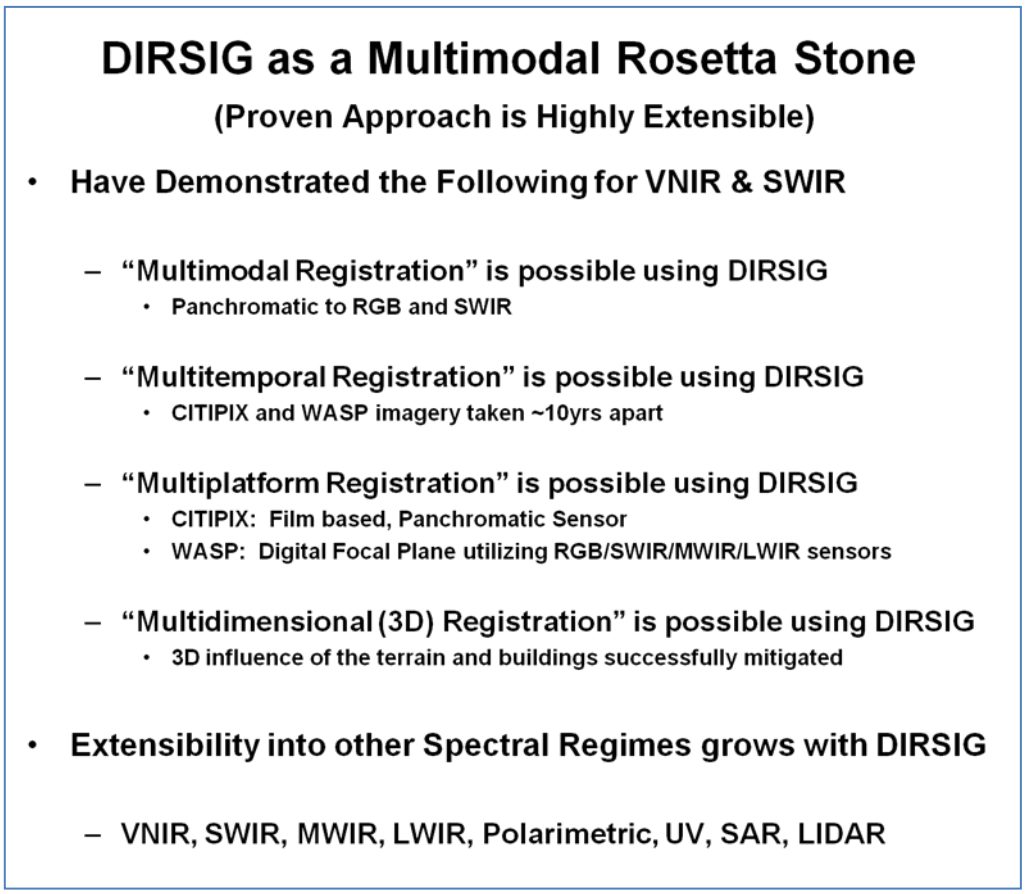


Figure 6-44 A summary of the DIRSIG Rosetta Stone strengths regarding multimodal image registration.

7 Relating Results in the World Coordinate System

The fundamental research into relating and combining sparse and faceted structure has just begun. With the recent commercial interest into SfM and LIDAR products, the ability to relate the resulting structural products with additional imagery modalities is ripe for research investment. Additionally, the ability to associate remotely sensed images within a GIS environment, similar to Section 3.1.2, provides the ability to mathematically relate the entire multi-view ensembles of camera locations, images and sparse structure (Chapter 4) to a global scene. The synergy of relating these multimodal image bundles and models, while having the ability to seamlessly interacting with them in a mathematical manner, will provide a venue for additional data fusion and derived product research.

Mathematically relating the resulting image bundles to the World Coordinate System (WCS) is depicted in Figure 7-1, where the 3D structure of the bundle is designated X and the same structure located within the WCS is X_o. Although the initial structure can be easily related w.r.t. the WCS once the proper transform parameters are recovered (Chapter 5) via a 3D Homography ($H_{4\times4}$), automatically correlating this structure to features within a GIS is nontrivial and is central to the basic research of this overall effort.

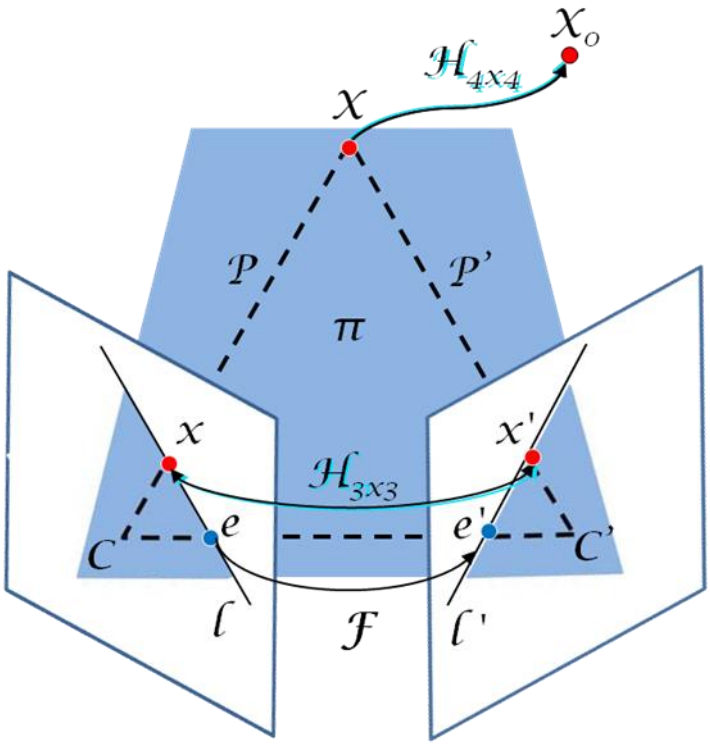


Figure 7-1 Relating the cameras, images, and structure to a World Coordinate System augments the mathematical relationships developed in Chapter 4, by combining it with the 3D Conformal techniques of Chapter 0 within a GIS construct.

In order to relate localized image bundles, site models, and collected imagery to the real world, it is first necessary to introduce a few additional mathematical techniques that will help relate these products within the WCS. Of key interest is incorporating the local 2D and 3D mathematical relationships within a more global 3D construct. To enable this, the epipolar geometry techniques introduces in Chapters 2-4, such as the Fundamental Matrix and 2D Homography will be extended to the WCS, while new concepts such as an epipolar plane π (Figure 7-1) will be introduced (Hartley and Zisserman 2004). It should be kept in mind that while F is inherently different than the homography H_π, which can be utilized to directly relate x and x' via an epipolar plane π,

$$x' = H_\pi x \tag{104}$$

we can still relate F and H_π via the following relationship.

$$F = [e']_\times H_\pi \tag{105}$$

This allows both the homography and the epipolar line or the fundamental matrix to constrain correspondences as noted below.

$$l' = [e']_\times H_\pi x = Fx \tag{106}$$

To help clarify the relationship between a planar homography H_π and the homography H, that we've been employing thus far, it is essential to realize that there is a perspectivity between the world plane point x_π, the first image plane $x = H_{1\pi}\, x_\pi$ and the second image plane $x' = H_{2\pi}\, x_\pi$. The composition of these two perspectivities is a homography (Hartley and Zisserman 2004)

$$x' = H_{2\pi} H_{1\pi}^{-1} x = H_\pi x = H_{3x3} x = Hx \tag{107}$$

Now we can relate, x to x' via H_{3x3}, x to X using P_{3x4}, and X to X_0 with H_{4x4}, by using the following additional equations (Hartley and Zisserman 2004)

Camera Projection Matrix

$$x = PX \tag{108}$$

3D Homography

$$X_0 = H_{4x4} X \tag{109}$$

Transformed Projection Matrix

$$P_0 = P H_{4x4}^{-1} \tag{110}$$

Local & Global Image Projection

$$P_0 X_0 = P H_{4x4}^{-1} H_{4x4} X = PX = x \tag{111}$$

It is interesting that the rather simple idea, of relating images to their corresponding world coordinates systems as expressed in Section 3.1.2, also offers the key to relating all of our desired modality and dimensionalities. If we can relate these bundles to a GIS environment via a 2D Homography H_{3x3}, then relating the GIS viewport image (as in Section 3.1.2) and the base image of the bundle has profound implications, because it will allow us to directly relate all of the datasets and have them globally referenced. This framework will allow the combination of Sparse/Dense Point Clouds, image bundles, multimodal images, and LIDAR datasets to be referenced and registered in the world coordinate system for truly integrated analysis and exploitation.

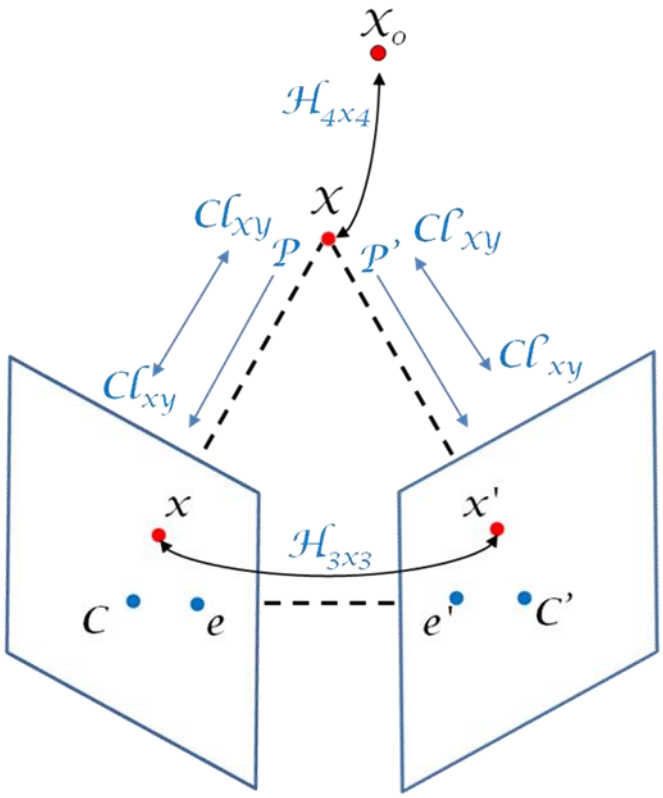


Figure 7-2 The relationships between the 2D/3D Homographies (H), Projection Matrix (P), and Colinearity Equations.

The graphic above depicts the various mathematical methods that can be utilized to relate various aspects of an image bundle. The reader should note the inclusion of the Colinearity equations, which have been a cornerstone of Photogrammetric analysis for decades.

Colinearity Eq x-component image proj.

$$x - x_0 = -f\left[\frac{m_{11}(X - X_L) + m_{12}(Y - Y_L) + m_{13}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)}\right] \quad \textbf{(112)}$$

Colinearity Eq y-component image proj.

$$y - y_0 = -f\left[\frac{m_{21}(X - X_L) + m_{22}(Y - Y_L) + m_{23}(Z - Z_L)}{m_{31}(X - X_L) + m_{32}(Y - Y_L) + m_{33}(Z - Z_L)}\right] \quad \textbf{(113)}$$

Colinearity Eq X-component World proj.

$$X - X_L = (Z - Z_L)\left[\frac{m_{11}(x - x_0) + m_{21}(y - y_0) + m_{31}(-f)}{m_{13}(x - x_0) + m_{23}(y - y_0) + m_{33}(-f)}\right] \quad \textbf{(114)}$$

Colinearity Eq Y-component World proj.

$$Y - Y_L = (Z - Z_L)\left[\frac{m_{12}(x - x_0) + m_{22}(y - y_0) + m_{32}(-f)}{m_{13}(x - x_0) + m_{23}(y - y_0) + m_{33}(-f)}\right] \quad \textbf{(115)}$$

By careful inspection, the similarities can be seen between the Colinearity equations of Equations (112) and (113) and the Camera Projection Matrix (122); here the simplified form of the IOP matrix (K) is utilized where $sk = 0$ and $f = \alpha_x = \alpha_y$ as in Chapter 11.

Camera Projection Matrix

$$P = KR[I \mid -t] \quad \textbf{(116)}$$

Proj Matrix Sub-Matrices

$$P = \begin{bmatrix}\alpha_x & sk & x_0\\ 0 & \alpha_y & y_0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}R_{11} & R_{12} & R_{13}\\ R_{21} & R_{22} & R_{23}\\ R_{31} & R_{32} & R_{33}\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & -T_x\\ 0 & 1 & 0 & -T_y\\ 0 & 0 & 1 & -T_z\end{bmatrix} \quad \textbf{(117)}$$

$$P = \begin{bmatrix}f & 0 & x_0\\ 0 & f & y_0\\ 0 & 0 & 1\end{bmatrix}\begin{bmatrix}R_{11} & R_{12} & R_{13}\\ R_{21} & R_{22} & R_{23}\\ R_{31} & R_{32} & R_{33}\end{bmatrix}\begin{bmatrix}1 & 0 & 0 & -T_x\\ 0 & 1 & 0 & -T_y\\ 0 & 0 & 1 & -T_z\end{bmatrix} \quad \textbf{(118)}$$

Projection Transform

$$\vec{x} = P\vec{X} = P\begin{bmatrix}X\\ Y\\ Z\\ 1\end{bmatrix} \quad \textbf{(119)}$$

$$\vec{x} = \begin{bmatrix} (fR_{11} + x_0R_{31})X + (fR_{12} + x_0R_{32})Y + (fR_{13} + x_0R_{33})Z - (fR_{11} + x_0R_{31})T_x - (fR_{12} + x_0R_{32})T_y - (fR_{13} + x_0R_{33})T_z \\ (fR_{21} + x_0R_{31})X + (fR_{22} + x_0R_{32})Y + (fR_{23} + x_0R_{33})Z - (fR_{21} + x_0R_{31})T_x - (fR_{22} + x_0R_{32})T_y - (fR_{23} + x_0R_{33})T_z \\ R_{31}X + R_{32}Y + R_{33}Z - R_{31}T_x - R_{32}T_y - R_{33}T_z \end{bmatrix} \quad \textbf{(120)}$$

Collect Terms

$$\vec{x} = \begin{bmatrix} x_0\left(R_{31}(X-T_x) + R_{32}(Y-T_y) + R_{33}(Z-T_z)\right) + f\left(R_{11}(X-T_x) + R_{12}(Y-T_y) + R_{13}(Z-T_z)\right) \\ y_0\left(R_{31}(X-T_x) + R_{32}(Y-T_y) + R_{33}(Z-T_z)\right) + f\left(R_{21}(X-T_x) + R_{22}(Y-T_y) + R_{23}(Z-T_z)\right) \\ R_{31}(X-T_x) + R_{32}(Y-T_y) + R_{33}(Z-T_z) \end{bmatrix} \quad \textbf{(121)}$$

Simplify to Collinearity Form

$$\vec{x} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = \begin{bmatrix} x_0 \\ y_0 \\ 0 \end{bmatrix} + \begin{bmatrix} f\left(R_{11}(X-T_x) + R_{12}(Y-T_y) + R_{13}(Z-T_z)\right) \\ f\left(R_{21}(X-T_x) + R_{22}(Y-T_y) + R_{23}(Z-T_z)\right) \\ R_{31}(X-T_x) + R_{32}(Y-T_y) + R_{33}(Z-T_z) \end{bmatrix} \quad \textbf{(122)}$$

Except for the sign convention (induced by the camera's distance, $\pm f$ from the focal plane), Equation (122) is equivalent to the Colinearity Equations (112) and (113) after division of the x/y components (rows 1 & 2) by the scaling component (row 3).

Although the Camera Projection Matrix has a compact form and is frequently utilized in computer vision to provide the 2D projected view of a 3D scene onto the focal plane, the "back-projection" matrix has the ability to relate the image location to an X-Y position in 3D space, a given distance (Z) from the camera and is equivalent to the remaining two Colinearity Equations. The proof of the matrix form of the back projection has been developed by the author as part of this dissertation proposal in Equations (123) through (132).

Projection Matrix

$$\vec{x} = P\vec{X} = KR[I \mid -t]\begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = KR\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \quad \textbf{(123)}$$

Rearrange

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = R^{-1}K^{-1}\vec{x} \quad \textbf{(124)}$$

$R^T=R^{-1}$

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = R^T K^{-1}\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \textbf{(125)}$$

Expand

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}^T \begin{bmatrix} -f & 0 & x_0 \\ 0 & -f & y_0 \\ 0 & 0 & 1 \end{bmatrix}^{-1} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \textbf{(126)}$$

Invert K

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} R_{11} & R_{21} & R_{31} \\ R_{12} & R_{22} & R_{32} \\ R_{13} & R_{23} & R_{33} \end{bmatrix} \begin{bmatrix} -1/f & 0 & x_0/f \\ 0 & -1/f & y_0/f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \quad \textbf{(127)}$$

Multiply

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} - \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = \begin{bmatrix} \frac{-\mathrm{R}_{11} * \mathrm{x}}{\mathrm{f}} - \frac{\mathrm{R}_{21} * \mathrm{y}}{\mathrm{f}} + \frac{\mathrm{R}_{11} * \mathrm{x}_0}{\mathrm{f}} + \frac{\mathrm{R}_{21} * \mathrm{y}_0}{\mathrm{f}} + \mathrm{R}_{31} \\ \frac{-\mathrm{R}_{12} * \mathrm{x}}{\mathrm{f}} - \frac{\mathrm{R}_{22} * \mathrm{y}}{\mathrm{f}} + \frac{\mathrm{R}_{12} * \mathrm{x}_0}{\mathrm{f}} + \frac{\mathrm{R}_{22} * \mathrm{y}_0}{\mathrm{f}} + \mathrm{R}_{32} \\ \frac{-\mathrm{R}_{13} * \mathrm{x}}{\mathrm{f}} - \frac{\mathrm{R}_{23} * \mathrm{y}}{\mathrm{f}} + \frac{\mathrm{R}_{13} * \mathrm{x}_0}{\mathrm{f}} + \frac{\mathrm{R}_{23} * \mathrm{y}_0}{\mathrm{f}} + \mathrm{R}_{33} \end{bmatrix} \quad \textbf{(128)}$$

Divide Row1 by Row3

$$X - T_x = (Z - T_z) \left[\frac{\frac{\mathrm{R}_{11} * \mathrm{x} + \mathrm{R}_{21} * \mathrm{y} - \mathrm{R}_{11} * \mathrm{x}_0 - \mathrm{R}_{21} * \mathrm{y}_0 - \mathrm{R}_{31} * \mathrm{f}}{-\mathrm{f}}}{\frac{\mathrm{R}_{13} * \mathrm{x} + \mathrm{R}_{23} * \mathrm{y} - \mathrm{R}_{13} * \mathrm{x}_0 - \mathrm{R}_{23} * \mathrm{y}_0 - \mathrm{R}_{33} * \mathrm{f}}{-\mathrm{f}}} \right] \quad \textbf{(129)}$$

Collinearity Eq X-component World proj.

$$X - T_x = (Z - T_z) \left[\frac{\mathrm{R}_{11}(\mathrm{x} - \mathrm{x}_0) + \mathrm{R}_{21}(\mathrm{y} - \mathrm{y}_0) + \mathrm{R}_{31}(-\mathrm{f})}{\mathrm{R}_{13}(\mathrm{x} - \mathrm{x}_0) + \mathrm{R}_{23}(\mathrm{y} - \mathrm{y}_0) + \mathrm{R}_{33}(-\mathrm{f})} \right] \quad \textbf{(130)}$$

Divide Row2 by Row3

$$Y - T_y = (Z - T_z) \left[\frac{\frac{\mathrm{R}_{12} * \mathrm{x} + \mathrm{R}_{22} * \mathrm{y} - \mathrm{R}_{12} * \mathrm{x}_0 - \mathrm{R}_{22} * \mathrm{y}_0 - \mathrm{R}_{32} * \mathrm{f}}{-\mathrm{f}}}{\frac{\mathrm{R}_{13} * \mathrm{x} + \mathrm{R}_{23} * \mathrm{y} - \mathrm{R}_{13} * \mathrm{x}_0 - \mathrm{R}_{23} * \mathrm{y}_0 - \mathrm{R}_{33} * \mathrm{f}}{-\mathrm{f}}} \right] \quad \textbf{(131)}$$

Collinearity Eq Y-component World proj.

$$Y - T_y = (Z - T_z) \left[\frac{\mathrm{R}_{12}(\mathrm{x} - \mathrm{x}_0) + \mathrm{R}_{22}(\mathrm{y} - \mathrm{y}_0) + \mathrm{R}_{32}(-\mathrm{f})}{\mathrm{R}_{13}(\mathrm{x} - \mathrm{x}_0) + \mathrm{R}_{23}(\mathrm{y} - \mathrm{y}_0) + \mathrm{R}_{33}(-\mathrm{f})} \right] \quad \textbf{(132)}$$

Thus, from Equation (124) it is easy to see that the following equation (133) represents the Back-Projection Transform and Equation (134) is the Back-Projection Matrix (B), where $\tilde{X}$ is the non-homogeneous form of the 3D location.

Back Projection Transform

$$\tilde{X} = R^{-1} K^{-1} \vec{x} + t \quad \textbf{(133)}$$

Back Projection Matrix

$$B = R^{-1}K^{-1}[I\ |t] \tag{134}$$

This mathematical proof should illuminate the connection between the Colinearity Equations traditionally utilized by the photogrammetrist and the Projection Matrix now commonly implemented by the computer vision community to relate 2D to 3D structure.

It should be of great concern to the modern Photogrammetry community to mathematically link their proven concepts and techniques to the growing field of computer vision, due to the incredible leveraging of ideas and techniques that can be accomplished. The author has great appreciation for the tremendous work done in both these fields to enable much of what was accomplished in this research.

8 Research Contributions

The following sections are designed to identify the specific contributions made by the author with the research included in this thesis.

8.1 Photogrammetric and Epipolar Geometry based Terrain Recovery

The technique developed by the author for 3D terrain recovery from imagery combines modern methods for 2D image registration, utilizing epipolar geometry constraints and outlier removal, and combines them with traditional photogrammetric approaches for 3D structure recovery. This has the added benefit of providing results where the recovered structure is represented in a global UTM coordinate system as opposed to a locally derived, relative structure. By utilizing the IOP and EOP of the camera, in this case WASP, the recovered image bundle and 3D point cloud can be utilized directly with modern GIS applications such as Google Earth to visualize the related imagery and 3D structure in the scene. The bullets below highlight some of the main contributions in this area of research:

- Indigenous CIS ability to derive structure from multiple images of a scene
- Improvements for GIS Applications (i.e. Planar outlier removal)
- Recovered structure and image bundle linked to WCS (global UTM)
- Allows direct comparison of SBA-SPC results with LIDAR and GIS Models
- Results can be used as a 3D Seed Model for DIRSIG Simulations similar to LIDAR

8.2 Constrained Conformal and Affine 3D Transformation

The author has developed innovative techniques to recover both the 3D Conformal and Affine Transformation parameters. Not only are these techniques essential for relating 3D rigid bodies, but, they help establish similar techniques for the recovery of constrained multidimensional transformation parameters through the use of SVD and QR decomposition. Although some of these techniques were borrowed from Multiple View Geometry (Hartley and Zisserman 2004), they have been reapplied here to address the purely 3D transformation problem (3D to 3D) that is not addressed in that text. The bullets below highlight some of the main contributions in this area of research:

- Development of a Constrained 3D Conformal and 3D Affine Transformation
- Application of 2D Multi-View Geometry Techniques to 3D
    - SVD Decomposition to recover embedded matrices
    - QR Decomposition to recover embedded matrices
    - Nonlinear Weighting to diminish unwanted terms

8.3 DIRSIG 3D Multimodal Registration

Although DIRSIG has been utilized for several pieces of fundamental research in the areas of multimodal analysis (spectral, polarimetric, LIDAR, and SAR), the author does not know of any attempts to use it to relate these modalities for registration. The key here is to accurately model the modalities, within a physics based environment, so that the synthetic results are similar enough to be utilized to automatically register with real data. Not only does this provide a “Rosetta Stone” to relate the datasets, it is also a testament to the ability of DIRSIG to replicate realistic results in these modalities and catalyzes the possibility of parallel growth and

research development. The bullets below highlight some of the main contributions in this area of research:

- Development of Techniques to use DIRSIG as a multimodal 'Rosetta Stone'
- Test the basic influences of a modeled scene for precise registration
    - Influences of the 3D structural model
    - Influences of the spectral attribution of model facets
- Development of both Hybrid and LIDAR-Direct modeling approaches within DIRSIG

8.4 Comprehensive Breadth – Multi-Dimensional/Modal Research

The author has not seen the comprehensive breadth of research into 3D multimodal registration covered in any one source, especially as it is applied to the area of remote sensing from aerospace platforms. Although several pieces of literature cover specific aspects of 3D multimodal registration, none of them cover the breadth of techniques and datasets that are covered here.

- 2D Image Registration in a 3D environment (H_{3x3})
- 2D Image Relationship to 3D Model (P_{3x4} and Colinearity Eqs.)
- 3D Structure to 3D Model Registration (H_{4x4})
- Combined structural and physical models accurate enough to register with real imagery

8.5 Suite of MATLAB Software Tools

A comprehensive table and flowchart of the various tools and applications that were developed in the process of completing this research will be delivered with the code shortly after the dissertation defense. However a few of the more important deliverables are highlighted below.

- AeroSynth Sparse Point Cloud Software Toolkit

- 2D Image Registration Toolkit that incorporates Epipolar Constraints
- LIDAR Processing Toolkit for Reading, Extracting, and Facetizing a Dense Point Cloud
- 3D Pose Estimation from Imagery code
- 3D Rigid Body Registration code

9 Summary

The research covered in this dissertation has focused on developing the essential mathematical foundation and techniques required to relate multimodal imagery data in 3D. This research has resulted in new tools/algorithm development and has improved techniques to accurately and efficiently relate datasets of interest to the remote sensing community.

By developing a 'model centric' approach to registration, it is possible to address both the influences of the 3D scene and the multimodal appearance of an image, which are currently the most challenging problems in the image registration arena. Geometric modeling of a scene allows mitigation of parallax, occlusion, and shadowing effects, while physical modeling (via DIRSIG), makes it possible to account for changes in appearance due to multimodal sensing effects. Utilized together, both the geometric and physical modeling of a scene allow automatic registration of images collected in different regions of the EMS, taken from various sensor-to-scene geometries and lighting conditions. This 'model centric' approach is a higher level of extrapolation than traditional 'image content' based approaches, in that it tries to maximize similarity of the image to the geometric/physical model, register the image to the projection of the model, and then use that mathematical relationship to correctly archive the image onto the model as a texture layer.

It is the hope of the author, that by providing several case studies exemplifying various aspects of data registration, that this research will provide utility in several areas of interest to the Center for Imaging Science at RIT and the remote sensing community in general. To augment this goal, the mathematical techniques researched here have being developed using MATLAB

code (The Mathworks, Inc. 2010) as modular functions for ease of application to a broad range of remote sensing registration problems and is included as a library of functions.

Finally, it is the hope of the author that this research has adequately addressed the broad goal of relating various types of multimodal imagery data in 3D and has provided the committee with credible results for accomplishing this task.

References

10 References

Blender Foundation. *Blender.* 2010. http://www.blender.org/ (accessed July 22, 2010).

Bourke, Paul. *Object Files - Wavefront (.obj) file specification.* July 29, 2010. http://local.wasp.uwa.edu.au/~pbourke/dataformats/obj/ (accessed July 29, 2010).

Chandrasekhar, A. *Point extraction and matching for registration of infrared astronomical images (Thesis).* Rochester, NY: Rochester Institute of Technology, 1999.

Cyganek, Boguslaw. "An Algorithm for the Computation of the Scene Geometry by the Log-Polar Area Matching Around Salient Points." *34th Conference on Current Trends in Theory and Practice of Computer Science.* New York: Springer, 2008. 222-233.

Davis, P. "Levenberg-Marquardt Methods and Nonlinear Estimation." *Siam News - Volume 26-6*, October 1993.

Delaunay, Boris. "Sur la sphere vide (On the empty sphere)." *Bulletin of the Academy of Sciences of the USSR, Class of Natural Science and Mathematics*, 1934: 793-800.

DeWitt, Bon A., and Paul R. Wolf. *Elements of Photogrammetry (with Applications in GIS).* 3rd. McGraw-Hill Higher Education, 2000.

Digital Imaging and Remote Sensing Laboratory. *The DIRSIG User's Manual.* Rochester, NY: Rochester Institute of Technology, 2006.

Drakos, Nikos, and Ross Moore. *Computer Based Learning Unit, University of Leeds.* September 18, 2007. http://fourier.eng.hmc.edu/e161/lectures/gradient/node12.html (accessed July 14, 2009).

Fan, X., H. Rhody, and E. Saber. "An algorithm for automated registration of maps and images based on feature detection and mutual information." *Electronic Imaging Conference.* San Jose, CA: SPIE, 2008.

Fiete, Robert, and Theodore Tantalo. "Comparison of SNR image Quality metrics for remote sensing systems." *Optical Engineering*, 2001: 574-585.

Fischler, Martin, and Robert Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with applications to Image Analysis and Automated Cartography." *Communications of the ACM, Volume24, Issue 6.* New York: ACM, 1981. 381-395.

Gartley, Michael, Adam Goodenough, Scott Brown, and Russel Kauffman. "A comparison of spatial sampling techniques enabling first principles modeling of a synthetic aperture RADAR imaging platform." *Algorithms for Synthetic Aperture Radar Imagery XVII, SPIE Vol. 7699.* SPIE, 2010.

GeoEye, Inc. *Imagery Sources.* 2010. http://www.geoeye.com/CorpSite/products/imagery-sources/Default.aspx (accessed July 22, 2010).

Gonzalez, R. C., and R. E. Woods. *Digital Image Processing, 3rd ed.* Upper Saddle River, NJ: Prentice Hall, 2007.

Google Earth. *Google Earth Homepage.* 2010. http://earth.google.com/ (accessed July 22, 2010).

Google Sketchup. *Google Sketchup Homepage.* 2009. http://sketchup.google.com/ (accessed July 13, 2009).

Gurram, Prudvi, Harvey Rhody, John Kerekes, Steve Lach, and Eli Saber. "3D scene reconstruction through a fusion of passive video and lidar imagery." *Proceedings of the Applied Imagery Pattern Recognition Workship.* Washington, DC: IEEE, 2007.

Haralick, R. M., H Joo, C. N. Lee, X. H. Zhuang, V. G. Vaidya, and M. B. Kim. "Pose Estimation From Corresponding Point Data." *IEEE TRANSACTIONS ON SYSTEMS MAN AND CYBERNETICS* 19 (1989): 1426-1446.

Harris, C., and M. Stephens. "A Combined Corner and Edge Detection." 1988. 147-151.

Hartley, R. I., and A. Zisserman. *Multiple View Geometry in Computer Vision.* Second. Cambridge University Press, ISBN: 0521540518, 2004.

Heikkila, J, and O Silven. "A four-step camera calibration procedure with implicit image correction." 1997. 1106-1112.

ITT Visual Information Solutions. *IDL Homepage.* 2008. http://www.ittvis.com/ProductServices/IDL.aspx (accessed July 13, 2009).

Kim, Sooyoung, Thomas Hinckley, and David Briggs. "Classifying tree species using structure and spectral data from LIDAR." *ASPRS/MAPPS 2009 Specialty Conference.* San Antonio: ASPRS, 2009.

Kodak Global Imaging. *Citipix Imagery.* 2008. http://www.library.ucsb.edu/citipix/ (accessed July 22, 2010).

Kraus, Karl, Ian Harley, and Stephen Kyle. *Photogrammetry, Geometry from Images and Laser Scans (2nd Ed.).* Berlin: Walter de Gruyter, 2007.

Kucera International Inc. *Kucera International Inc.* 2010. http://www.kucerainternational.com/ (accessed July 22, 2010).

Lindeberg, T. "Scale-space theory: A basic tool for analysing structures at different scales." *Journal of Applied Statistics*, 1994: 21(2):224-270.

Lourakis, M. I. A., and A. A. Argyros. "SBA: A Software Package for Generic Sparse Bundle Adjustment." *ACM TRANSACTIONS ON MATHEMATICAL SOFTWARE* 36 (2009).

Lourakis, M.I.A., and A. A. Argyros. *sba : A Generic Sparse Bundle Adjustment C/C++ Package Based on the Levenberg-Marquardt Algorithm.* Aug 31, 2010. http://www.ics.forth.gr/~lourakis/sba/ (accessed July 22, 2010).

Lourakis, M.I.A., and A.A. Argyros. "The Design and Implementation of a Generic Sparse Bundle Adjustment Software Package Based on the Levenberg-Marquardt Algorithm." Institute of Computer Science - FORTH, 2004.

Lowe, D. G. "Distinctive image features from scale-invariant keypoints." *INTERNATIONAL JOURNAL OF COMPUTER VISION* 60 (2004): 91-110.

Lowe, David. *Demo Software: SIFT Keypoint Detector.* July 2005. http://www.cs.ubc.ca/~lowe/keypoints/ (accessed Aug 2, 2010).

Ma, Yi, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An Invitation to 3-D Vision, From Images to Geometric Models (2nd Edition).* New York: Springer, 2006.

Madsen, K., H. B. Nielsen, and O. Tingleff. "Methods for Non-Linear Least Squares Problems (2nd Edition)." *Technical University of Denmark Public Document Server.* April 2004. http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/3215/pdf/imm3215.pdf (accessed July 13, 2009).

Microsoft Corporation. *Microsoft Bing Maps.* 2010. http://www.bing.com/maps/ (accessed July 22, 2010).

—. *Photosynth.* 2010. http://photosynth.net/ (accessed July 22, 2010).

Mikolajczyk, K., and C. Schmid. "A Performance Evaluation of Local Descriptors." *IEEE Trans. Pattern Anal. Mach. Intell.*, 2005: 1615-1630.

Mikolajczyk, Krystian. *Detection of local features invariant to affine transformation (Thesis).* Sant Ismier, FR: Institut National Polytechnique de Grenoble, 2002.

Moreels, P., and P. Perona. "Evaluation of Features Detectors and Descriptors based on 3D Objects." *International Journal of Computer Vision*, 2006: 263-284.

Nilosek, David, Karl Walli, Carl Salvaggio, and John Schott. "AeroSynth: Aerial scene synthesis from images." *SIGGRAPH 2009, 36th International Conference and Exhibition on Computer Graphics and Interactive Techniques.* New Orleans: SIGGRAPH, 2009.

Pictometry, Corporation Int. *Pictometry Homepage.* 2010. http://www.pictometry.com/home/home.shtml (accessed July 22, 2010).

Pisa, University of. *MeshLab.* May 6, 2010. http://meshlab.sourceforge.net/ (accessed July 16, 2010).

Pollefeys, M, et al. "Visual modeling with a hand-held camera." *INTERNATIONAL JOURNAL OF COMPUTER VISION* 59 (2004): 207-232.

Press, William H., Brian P. Flannery, Saul A. Teukolsky, and William T. Vetterling. *Numerical Recipes in C : The Art of Scientific Computing.* Cambridge University Press, 1992.

Ranganathan, Ananth. "The Levenberg-Marquardt Algorithm." *The Levenberg-Marquardt Algorithm.* June 8, 2004. http://cronos.rutgers.edu/~meer/TEACH/ADD/ananth.pdf (accessed July 30, 2010).

Rhody, H. "Levenberg-Marquardt Minimization." *Class Notes.* Rochester, New York: Rochester Institute of Technology, February 10, 2009.

Rhody, Harvey. "Notes on Automatically Relating 3D Objects." Rochester, NY, July 22, 2010.

Scanlan, Neil. "Comparative performance analysis of texture characterization models in DIRSIG." *Master's Thesis.* Rochester, NY: Rochester Institute of Technology, August 2003.

Schott, John R. *Remote Sensing: The Image Chain Approach.* New York: Oxford University Press, 2007.

Schowengerdt, R. *Remote sensing, models and methods for image processing (3rd Edition).* New York: Academic Press, 2007.

Seedahmed, Gamal H. "Direct retrieval of exterior orientation parameters using a 2D projective transformation." *PHOTOGRAMMETRIC RECORD* 21 (2006): 211-231.

Snavely, Noah. *Bundler.* April 10, 2010. http://phototour.cs.washington.edu/bundler/ (accessed July 22, 2010).

Snavely, Noah, Steven M. Seitz, and Richard Szeliski. "Photo tourism: Exploring photo collections in 3D." *ACM TRANSACTIONS ON GRAPHICS* 25 (2006): 835-846.

The Mathworks, Inc. *MATLAB Hompage.* 2010. http://www.mathworks.com/ (accessed July 22, 2010).

Van Nevel, Alan. "Image registration: a key element for information processing." *Algorithms and Systems for Optical Information Processing V, SPIE Vol. 4471.* SPIE, 2001. 190-200.

Walli, Karl. *Multisensor Image Registration utilizing the LoG Filter and FWT.* Rochester, NY: Rochester Institute of Technology, 2003.

Walli, Karl, and Harvey Rhody. "Automated Image Registration to 3-D Scene Models." *Applied Imagery Pattern Recognition Workshop.* Washington, D.C.: IEEE Computer Society, 2008.

Walli, Karl, David Nilosek, John Schott, and Carl Salvaggio. "Airborne Synthetic scene generation (AeroSynth) ." *ASPRS/MAPPS 2009 Fall Conference, Digital Mapping - From Elevation to Information.* San Antonio: ASPRS, 2009.

Wolberg, G. *Digital Image Warping.* Los Alamito, CA: IEEE Computer Society Press, 1990.

Zhang, Zhengyou. "Iterative point matching for registration of free-form curves and surfaces." *International Journal of Computer Vision*, 1992: 119-152.

Zhang, ZY. "A flexible new technique for camera calibration." *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* 22 (2000): 1330-1334.

Appendices

11 APPENDIX A - Camera Calibration

In order to fully understand the capabilities and limitations in relating multimodal datasets, it is important to mathematically model the sensor's interaction with the world. For most sensors, this involves some type of camera calibration to determine its external 3D location/orientation and internal characteristics. Some of the basic techniques to accomplish this task are covered below.

11.1 The Camera External Orientation Parameters (EOPs)

This section provides the mathematical representation of a camera's external orientation parameters and their application to a homogeneous coordinate system. These parameters are characterized by the local or global location and orientation of the camera during the image acquisition. The position vector is contained in a 3-vector such in Equation (135) below.

3D Translation Vector

$$T_{3\times1} = \begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} \quad \textbf{(135)}$$

The 3D Rotation Matrix can be applied individually (136) as roll (ω), pitch (φ), and yaw/heading (κ) or as a composite transform as shown below (137); the order of axes rotation is important. If we assign $c = cos$ and $s = sin$, then the rotation matrices obtain the following form.

3D Rotation Matrix

$$R = R_{\kappa}R_{\varphi}R_{\omega} = \begin{bmatrix} c\kappa & -s\kappa & 0 \\ s\kappa & c\kappa & 0 \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} c\varphi & 0 & s\varphi \\ 0 & 1 & 0 \\ -s\varphi & 0 & c\varphi \end{bmatrix}\begin{bmatrix} 1 & 0 & 0 \\ 0 & c\omega & -s\omega \\ 0 & s\omega & c\omega \end{bmatrix} \quad \textbf{(136)}$$

3D Composite Rotation Transform

$$R = \begin{bmatrix} c\varphi c\kappa & (s\omega s\varphi c\kappa - c\omega s\kappa) & (c\omega s\varphi c\kappa + s\omega s\kappa) \\ c\varphi s\kappa & (s\omega s\varphi s\kappa + c\omega c\kappa) & (c\omega s\varphi s\kappa - s\omega c\kappa) \\ -s\varphi & s\omega c\varphi & c\omega c\varphi \end{bmatrix} \quad \textbf{(137)}$$

$$= \begin{bmatrix} R_{11} & R_{12} & R_{13} \\ R_{21} & R_{22} & R_{23} \\ R_{31} & R_{32} & R_{33} \end{bmatrix}$$

These vectors and matrices can then be placed into a homogeneous 3D Camera matrix for manipulation in graphical environments as seen in Equation (138).

3D Camera Description

$$H_{cam} = \begin{bmatrix} R_{11} & R_{12} & R_{13} & T_x \\ R_{21} & R_{22} & R_{23} & T_y \\ R_{31} & R_{32} & R_{33} & T_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \textbf{(138)}$$

11.2 The Camera Interior Orientation Parameters (IOPs)

This section provides a brief summary of the general internal calibration parameters for a basic pinhole camera, where $K$is the used to represent the internal calibration matrix.

Internal Calibration Matrix

$$K = \begin{bmatrix} \alpha_x & sk & x_0 \\ 0 & \alpha_y & y_0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(139)}$$

In this description, α_x and α_y represent the focal length of the camera in terms of pixel dimensions (x and y pixel pitch) and when combined as a ratio, give the sensor aspect ratio. Here, sk is the skew and x_0 and y_0 represent the focal plane's principle point. When $sk = 0$ and the image principal points are located at the origin $[x_0, y_0] = [0, 0]$, then

Internal Calibration Matrix

$$K = \begin{bmatrix} \alpha_x & 0 & 0 \\ 0 & \alpha_y & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(140)}$$

Since $\alpha_x = f_x\, m_x$ and $\alpha_y = f_y\, m_x$ where m is the number of pixels per unit length [m] ($m = (pixel\ pitch)^{-1}$) and f is the focal length [m] along the x and y axis, then this can be simplified to the following form when the pixels are square.

Internal Calibration Matrix

$$K = \begin{bmatrix} \alpha & 0 & 0 \\ 0 & \alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(141)}$$

Often, our linear estimate will have very small variations from a perfectly square pixel. In this case, the overall scale factor can be constrained to square pixels by averaging the results of α_x and α_y (Snavely, Seitz and Szeliski, Photo tourism: Exploring photo collections in 3D 2006), so that

Internal Calibration Matrix

$$K = \begin{bmatrix} \frac{\alpha_x + \alpha_y}{2} & 0 & 0 \\ 0 & \frac{\alpha_x + \alpha_y}{2} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad \textbf{(142)}$$

11.3 Radial Lens Distortion Parameters

Often camera calibration discussions will be completely devoid of discussing the important topic of radial lens distortion. While many of today's low end cameras are affected by this aberration, it is many times not readily visible in the resulting images. However, when trying to recover 3D structure from imagery by utilizing techniques like SBA (as in Chapter 4), correcting for radial lens distortion becomes fundamentally important.

This is especially true when imaging in the longer wavelengths of the infrared spectrum, where calibration becomes critical to relate multimodal imagery, since the index of refraction of these

camera lenses are often more dense than traditional "visible" light cameras. This is because the light rays are bent disproportionately from the image center and a radially increasing effect off the optical axis is manifested within the imagery as "barrel distortion". Once properly corrected, the image will have a noticeable concave exterior shape that is referred to as the "pin-cushioning" effect.

Zhang notes that most camera distortion is dominated by radial effects and that the first radial distortion term is of most significance (Z. Zhang 2000). Here we will utilize the first two radial distortion terms, where we solve for those terms using point correspondences. Zhang's concise treatment of radial distortion correction follows. Let $(\breve{x}, \breve{y})$ represent the radially distorted (normalized) image coordinates and (x,y) represent the corrected locations. Then the radial distortion can be expressed as

Radial Distortion x-component

$$\breve{x} = x + x\left[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2\right] \quad \textbf{(143)}$$

Radial Distortion y-component

$$\breve{y} = y + y\left[k_1(x^2 + y^2) + k_2(x^2 + y^2)^2\right] \quad \textbf{(144)}$$

where, k_1 and k_2 represent the two radial distortion coefficients that were previously mentioned. Since the image principal point (x_0, y_0) is the origin of the radial distortion, the above equations can be modified to more accurately index the pixels from an undistorted (u, v) location. The radially distorted coordinates $(\breve{u}, \breve{v})$, can then be derived from the $(\breve{x}, \breve{y})$ locations through the following relationships.

Radial Distortion $\breve{u}$-component

$$\breve{u} = x_0 + \alpha_x \breve{x} + sk\breve{y} \quad \textbf{(145)}$$

Radial Distortion $\breve{v}$-component

$$\breve{v} = y_0 + \alpha_y \breve{y} \tag{146}$$

If we assume skew is insignificant $(sk \cong 0)$, we can represent this new, centered coordinate system, in the (u, v) plane as

Radial Distortion $\breve{u}$-component

$$\breve{u} = u_i + (u_i - x_0)\left[k_1(x_i^2 + y_i^2) + k_2(x_i^2 + y_i^2)^2\right] \tag{147}$$

Radial Distortion $\breve{v}$-component

$$\breve{v} = v_i + (v_i - y_0)\left[k_1(x_i^2 + y_i^2) + k_2(x_i^2 + y_i^2)^2\right] \tag{148}$$

Where (u_i, v_i) are the undistorted pixel locations of our known model points X_i, projected through a pinhole camera model. In order to arrive at a linear estimate for the radial distortion coefficients, we can use the following DLT technique, which is explained in greater detail in the following Appendix (12).

Linear Estimate

$$\begin{bmatrix} \breve{u}_1 - u_1 \\ \breve{v}_1 - v_1 \\ \vdots \\ \breve{u}_i - u_i \\ \breve{v}_i - v_i \end{bmatrix} = \begin{bmatrix} (u_1 - x_0)(x_1^2 + y_1^2) & (u_1 - x_0)(x_1^2 + y_1^2)^2 \\ (v_1 - y_0)(x_1^2 + y_1^2) & (v_1 - y_0)(x_1^2 + y_1^2)^2 \\ \vdots & \vdots \\ (u_i - x_0)(x_i^2 + y_i^2) & (u_i - x_0)(x_i^2 + y_i^2)^2 \\ (v_i - y_0)(x_i^2 + y_i^2) & (v_i - y_0)(x_i^2 + y_i^2)^2 \end{bmatrix} \begin{bmatrix} k_1 \\ k_2 \end{bmatrix} \tag{149}$$

where the image correspondences can either be the related points in another corrected image, projected model points, or straight line estimated locations within the same image. Here, we can again utilize the Pseudo-Inverse to provide a solution to our linear least squares estimate.

Simplified Matrix Notation

$$d = Dk \tag{150}$$

Pseudo Inverse Solution to LLS

$$k = \left(D^T D\right)^{-1} D^T d \tag{151}$$

Finally, our Maximum Likelihood Estimate, utilizing a nonlinear optimization technique like LMA (Chapter 13-Appendix C), will be minimized against the following cost function that directly incorporates the two radial distortion coefficients.

Nonlinear Minimization Equation

$$\sum_{i=1}^{n} \left\| \breve{x}_i - \breve{X}_i(K, k_1, k_2, R, t, X_i) \right\|^2 \tag{152}$$

where $\breve{X}_i$ is the radially distorted and transformed model location X_i, and $\breve{x}_i$ is the corresponding location within the distorted image.

12 APPENDIX B - Linear Estimation

Most of the techniques shown here, for relating imagery and models, require nonlinear methods to obtain an accurate solution. However, it is often useful to seed these methods with a linear estimate that gets them within the capture range of the global minimum to help avoid getting "trapped" within local valleys of the solution space.

12.1 The Projection Matrix Revisited

Although the Pseudo-Inverse can be utilized to provide a Linear Least Squares solution for square matrices (25), it is not suitable for some applications. In particular, the solutions for resectioning and SBA (Chapters 3 & 4), require a 3x4 matrix of coefficients. Utilizing homogeneous coordinate systems to represent both the 2D image coordinates and the 3D model points result in the following equations (Hartley and Zisserman 2004).

Projection Matrix Simplified

$$\boldsymbol{x}_i = \boldsymbol{P}\boldsymbol{X}_i \tag{153}$$

Projection Matrix Expanded

$$\begin{bmatrix} x_1 & x_2 & \cdots x_i \\ y_1 & y_2 & \cdots y_i \\ 1 & 1 & \cdots 1 \end{bmatrix} = \begin{bmatrix} P_{11} & P_{12} & P_{13} & P_{14} \\ P_{21} & P_{22} & P_{23} & P_{24} \\ P_{31} & P_{32} & P_{33} & P_{34} \end{bmatrix} \begin{bmatrix} X_1 & X_2 & \cdots X_i \\ Y_1 & Y_2 & \cdots Y_i \\ Z_1 & Z_1 & \cdots Z_i \\ 1 & 1 & \cdots 1 \end{bmatrix} \tag{154}$$

One alternative to solve this equation is to utilize the vector cross product, where P_1^T is the transposed 1st row of the projection matrix.

Cross Product Solution

$$\boldsymbol{x}_i \times \boldsymbol{P}\boldsymbol{X}_i = 0 \tag{155}$$

Expanded

$$\boldsymbol{x}_i \times \boldsymbol{P}\boldsymbol{X}_i = \begin{pmatrix} y_i P_3^T X_i - P_2^T X_i \\ P_1^T X_i - x_i P_3^T X_i \\ x_i P_2^T X_i - y_i P_1^T X_i \end{pmatrix} = 0 \quad \textbf{(156)}$$

12.2 The Direct Linear Transform (DLT)

The cross product can then be expressed in the following form (157); which has linearly dependent equations and can then be reduced due to (158). This cross product approach has made the equations linear in the unknowns (P) and for this reason is commonly called the Direct Linear Transform (DLT).

DLT Derived from Cross Product

$$\begin{bmatrix} 0 & -\boldsymbol{X}_i^T & y_i \boldsymbol{X}_i^T \\ \boldsymbol{X}_i^T & 0 & -x_i \boldsymbol{X}_i^T \\ -y_i \boldsymbol{X}_i^T & x_i \boldsymbol{X}_i^T & 0 \end{bmatrix} \begin{bmatrix} P^{1T} \\ P^{2T} \\ P^{3T} \end{bmatrix} = 0 \quad \textbf{(157)}$$

Reduced DLT

$$\begin{bmatrix} 0 & -\boldsymbol{X}_i^T & y_i \boldsymbol{X}_i^T \\ \boldsymbol{X}_i^T & 0 & -x_i \boldsymbol{X}_i^T \end{bmatrix} \begin{bmatrix} P^{1T} \\ P^{2T} \\ P^{3T} \end{bmatrix} = 0 \quad \textbf{(158)}$$

In expanded form (for clarity) this equation takes on the following form (Heikkila and Silven 1997),

Expanded DLT

$$\begin{bmatrix} X_1 & Y_1 & Z_1 & 1 & 0 & 0 & 0 & 0 & -X_1x_1 & -Y_1x_1 & -Z_1x_1 & -x_1 \\ 0 & 0 & 0 & 0 & X_1 & Y_1 & Z_1 & 1 & -X_1y_1 & -Y_1y_1 & -Z_1y_1 & -y_1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_2 & Y_2 & Z_2 & 1 & 0 & 0 & 0 & 0 & -X_2x_2 & -Y_2x_2 & -Z_2x_2 & -x_2 \\ 0 & 0 & 0 & 0 & X_2 & Y_2 & Z_2 & 1 & -X_2y_2 & -Y_2y_2 & -Z_2y_2 & -y_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ X_i & Y_i & Z_i & 1 & 0 & 0 & 0 & 0 & -X_ix_i & -Y_ix_i & -Z_ix_i & -x_i \\ 0 & 0 & 0 & 0 & X_i & Y_i & Z_i & 1 & -X_iy_i & -Y_iy_i & -Z_iy_i & -y_i \end{bmatrix} \begin{bmatrix} P_{11} \\ P_{12} \\ P_{13} \\ P_{14} \\ P_{21} \\ P_{22} \\ P_{23} \\ P_{24} \\ P_{31} \\ P_{32} \\ P_{33} \\ P_{34} \end{bmatrix} = 0 \quad \textbf{(159)}$$

A popular and proven way to solve this equation and obtain the P coefficients is to use a robust technique like Singular Value Decomposition (SVD). A solution of $AP = 0$, subject to $\|P\| = 1$, is obtained from the singular vector of A corresponding to the smallest singular value (Hartley and Zisserman 2004). This equates to the last column of V if the diagonal values of D are in descending order. Below is a representation of the factorized form of A, where U and V are orthogonal matrices and D is a non-negative diagonal matrix.

Once the P vector is solved for in this form, it should be reshaped back into a 3x4 matrix (as in Eq. (154)) for implementation as a camera projection matrix. This will then allow projection of the model points onto the 2D space as viewed from a camera with the location and orientation embedded within the camera matrix P. At this point the linear estimate, provided by the DLT algorithm, can be fed into a nonlinear solver (such as Levenberg-Marquardt) as the initial starting point for a more precise iterative solution (covered in the next Appendix, Chapter 13).

13 APPENDIX C - Nonlinear Estimation

The Many of the problems presented in this research cannot be solved by linear methods alone. In these cases, it is necessary to apply non-linear estimation techniques to provide accurate solutions. Such real world problems as the resectioning of images to models and the Bundle Adjustment (BA) of multiple images, to reconstruct 3D structure, both require nonlinear minimization solutions. In fact, for BA, these solutions often depend on calculating the interaction of several thousand variables simultaneously. Due to its stability and speed of convergence, the Levenberg –Marquardt Algorithm (LMA) is currently the most popular approach to solve these challenging problems.

When utilizing LMA, the computational challenge is to minimize a given cost function. For applications such as resectioning and BA, this cost function is defined as the sum of the squared error between image points (actual data) and projected 3D model points (predicted values) dictated by the current set of parameter. The mathematical construct and implementation of the LMA are covered below.

13.1 The Levenberg-Marquardt Algorithm

The LMA is a hybrid approach to nonlinear estimation that interpolates between the Gauss-Newton algorithm (inverse Hessian) and the method of steepest (gradient) descent. When the current solution is far from the correct one, the algorithm behaves like a steepest descent method: slow, but guaranteed to converge. When the current solution is close to the correct solution, it becomes a Gauss-Newton method (Lourakis & Argyros 2004). Additionally, the

practical reliability of the method—the ability to converge promptly from a wider range of initial guesses than other typical methods—is a factor in its continued popularity (Davis 1993).

The following mathematical summary of the LMA is drafted primarily from Lourakis and Argyros (Lourakis & Argyros 2004) and Rhody (H. Rhody 2009). Let f be a function that maps a vector of parameters p to an estimated measurement vector $\hat{x}$.

Parameter Vector

$$p = [p_1 \quad p_2 \quad \cdots \quad p_m]^T \tag{160}$$

Functional Parameter Mapping

$$f(p) = \hat{x} \tag{161}$$

Now, we can define the difference between the actual measurement x and the estimate $\hat{x}$ as the residual, ε.

Residual Error

$$\varepsilon = x_m - \hat{x}_m \tag{162}$$

Expanded

$$\varepsilon = [(x_1 - \hat{x}_1), \quad (x_2 - \hat{x}_2), \quad \cdots, \quad (x_m - \hat{x}_m)] \tag{163}$$

Where, the Mean Squared Error (MSE) is,

Mean Square Error

$$MSE = \frac{1}{2}\sum_{i=1}^{n}(x_m - \hat{x}_m)^2 = \frac{1}{2}\|\varepsilon\|^2 = \frac{1}{2}\varepsilon^T\varepsilon \tag{164}$$

The basis of the LMA is a linear approximation to f in the neighborhood of p. For a small change in parameter space $\|\delta_p\|$, a Taylor series expansion leads to the following approximation.

Taylor Series Approx.

$$f(p+\delta_p) \approx f(p) + J\delta_p \tag{165}$$

where J is the Jacobian matrix of the function; which is the partial derivative of the function's predictions with respect to each of its parameters.

Jacobian

$$J = \frac{\partial f_i(p)}{\partial p_j} \tag{166}$$

Jacobian Matrix

$$J = \begin{bmatrix} \frac{\partial f_1(p)}{\partial p_1} & \frac{\partial f_1(p)}{\partial p_2} & \cdots & \frac{\partial f_1(p)}{\partial p_m} \\ \frac{\partial f_2(p)}{\partial p_1} & \frac{\partial f_2(p)}{\partial p_2} & & \frac{\partial f_2(p)}{\partial p_m} \\ \vdots & & \ddots & \vdots \\ \frac{\partial f_n(p)}{\partial p_1} & \frac{\partial f_n(p)}{\partial p_2} & \cdots & \frac{\partial f_n(p)}{\partial p_m} \end{bmatrix} \tag{167}$$

So that LMA can iterate toward a minimum, we must find a δ_p that minimizes the distance between our data measurement x and the new estimate $\hat{x}$, which is now $f(p+\delta p)$, where

Minimize Distance

$$\|x - f(p+\delta_p)\| \approx \|x - f(p) - J\delta_p\| = \|\varepsilon - J\delta_p\| \tag{168}$$

Our desired δ_p, is the solution to a linear least-squares problem, since, the minimum is attained when $J\delta_p - \varepsilon$ is orthogonal to the column space of J.

LLS Solution

$$J^T(J\delta_p - \varepsilon) = 0 \tag{169}$$

It follows that δ_p can now be considered a solution to an augmented form of the "normal equations" since,

LLS Solution

$$J^T J\delta_p = J^T\varepsilon \tag{170}$$

One solution for δ_p is through the use of the pseudo-inverse (Section 2.3),

Pseudo-Inverse Solution

$$\delta_p = \left(J^T J\right)^{-1} J^T \varepsilon = J^{\dagger} \varepsilon \tag{171}$$

In the case of BA, the δ_p solution is incorporated into the current iteration of the Camera Projection Matrix (P) and the residuals are recalculated and analyzed to determine if the total projected error has increased or decreased w.r.t. the following minimization function.

Projection Minimization Function

$$\sum_i d(x_i, PX_i)^2 \tag{172}$$

Expanded Minimization Function

$$\sum_{i=1}^{n} \left\| x_i - \widehat{X}_i(K, R, t, X_i) \right\|^2 \tag{173}$$

The LMA incorporates a unique damping scheme to minimize this error. If we analyze the normal equation representation and replace $J^T J$ with N:

Normal Equation Substitution

$$N\delta_p = J^T \varepsilon \tag{174}$$

Where the off diagonal elements of N are the same as the corresponding elements of $J^T J$, except that the diagonal elements have an added damping term μ, which explains the "augmentation" of the normal equations. Levenberg's contribution was to add this damping term.

Normal Equation Augmentation

$$N = \mu \cdot I + J^T J \tag{175}$$

Unfortunately, when the damping factor is large, then N^{-1} is not used at all, since δ_p approaches zero. Marquardt independently realized that if the identity matrix was replaced with the diagonal of the Hessian matrix, this could be avoided (Ranganathan 2004). Since the Hessian can be approximated by $J^T J$, as described below.

Hessian Diagonal Replacement

$$N = \mu \cdot I \cdot \left(J^T J\right) + J^T J \tag{176}$$

Hessian Notation

$$N = \mu\big(diag(H)\big) + H \tag{177}$$

Here, it is important to discuss some important properties of the Hessian matrix H, since it can be utilized to help determine the curvature of the nonlinear surface (the curvature matrix of a function is defined as $\frac{H}{2}$).

Hessian Surface Curvature

$$H = \nabla^2 f(p) = J^T J + \sum_{i=1}^{n} \varepsilon_i \nabla^2 \varepsilon_i \tag{178}$$

The last term can be ignored if the curvature of the surface is flat, or the residual error is approximately a linear function of P, or if the residual error is small.

Hessian Surface Curvature Approximation

$$H = \nabla^2 f(p) \approx J^T J \tag{179}$$

When using LMA, the damping term is adjusted at each iteration, to ensure a reduction in the residual error ε. One of Marquardt's insights was that the components of the Hessian matrix, even if they are not usable in any precise fashion, give some information about the order-of-magnitude scale of the nonlinear problem (Press, et al. 1992). This can help us understand the curvature of the parameter function at the current location.

When damping is set to a large value, the N matrix is diagonally dominant and the LMA update step δ_p is near the steepest descent direction and the magnitude of δ_p is reduced. Damping also handles situations where the Jacobian is rank deficient and $J^T J$ is therefore singular. In this way, LMA can defensively navigate a region of parameter space in which the model is

highly nonlinear (Lourakis & Argyros 2004). If the damping is small, LMA approximates the exact quadratic step appropriate for a fully linear problem, since the damping function disappears, thus becoming the Gauss-Newton method.

LM is adaptive because it controls its own damping: it raises the damping if a step fails to reduce ε; otherwise it reduces the damping. In this way LMA is able to alternate between a slow descent approach when far from the minimum and a fast convergence when it's in the neighborhood of the minimum. The LMA can be made to terminate when the magnitude of the gradient drops below a certain threshold (bottom of a valley), the relative change in the magnitude of the residual drops below a threshold, or a given number of iterations is complete.

It is important to note that one of Marquardt's improvements ensures that the detection of a local minimum of the cost function is not forced at each step. His subtle adjustment in the angle at which the method moves downhill provides quicker convergence because it avoids the steepest decent propensity to zigzag along a narrow valley, crossing and re-crossing the minimum before it reaches the bottom (Davis 1993).

14 APPENDIX D - Epipolar Techniques for Recovering Sparse Models

Unlike Section 4.2, in this appendix we will assume that we do not know the camera EOP/IOP or the real world coordinates of the image point correspondences. In fact, the final results of this process will only provide a relative SPC that is self-consistent with the image matches and derived camera parameters, not an absolute world coordinate solution. However, the power of this generalized solution is evident by the current popularity of applications such as PhotoSynth (Microsoft Corporation 2010) that provide this localized and sparse representation of the imaged scene without ever knowing many of the initial camera IOPs and EOPs.

14.1 Approach

The basic approach inherent to this technique is to relate images using invariant features and then utilize these correspondences and epipolar relationships to derive the relative relationships between the images and the imaged scene. Since the derived relationships and sparse structure are all relative to each other, in a localized coordinate system, this can be accomplished with little to no knowledge of the cameras and their positions.

Because of these initial conditions, there are two critical tasks that must be addressed in this appendix. First we must develop an estimate of the internal and external parameters for each camera. Second, we must provide estimates of the 3D locations for each of our point correspondences. This is a somewhat challenging task, due to the fact that there are normally 11 parameters for each of m cameras and 3 parameters for each of n 3D points, thus requiring $11 \times m + 3 \times n$ total parameter estimates.

14.2 Develop a Linear Estimate of the Camera Parameters

Of course, the number of camera parameters is often dependent on the number of assumptions that are being considered and can be as few as 5 or as many as 15 when solving for the radial distortion coefficients (152). Often the 12 parameters included in the projection matrix are solved for, due to the ease of minimization against this function within the LMA.

As noted by Wolf and Dewitt (DeWitt and Wolf 2000), for some near-nadir imaging cases, both the pitch (ϕ) and roll (ω) of the aircraft can be assumed as negligible for initial estimating purposes. Additionally, both of the skew parameters (sk_x & sk_y) and the principle point locations (x_0 & y_0) can be assumed equivalent to zero for most current framing sensors (Snavely, Seitz and Szeliski, Photo tourism: Exploring photo collections in 3D 2006). Additionally for framing sensors, the average of the two scaling parameters can be assumed as equivalent to the focal length (f) as indicated below.

$$f = \frac{(\alpha_x + \alpha_y)}{2} \quad \textbf{(180)}$$

So, for the near-nadir imaging case, a minimal set of 5 parameters, 4 EOPs and 1 IOP $(X_L, Y_L, X_L, \kappa, f)$ require initial linear estimates, where (X_L, Y_L, Z_L) represent the camera lens global location, kappa (k) represents the heading angle, and (f) is the camera's focal length. However, the 7 parameter set (that includes Omega and Phi) should be utilized for minimization when solving the nonlinear case. For more information on the internal camera calibration parameters please reference Chapter 11.

14.3 Develop a Linear Estimate of the 3D Points

The following three steps are utilized to estimate the initial guess at the 3D point location isolated from the correspondences in each image:

1. Derive the Fundamental Matrix F
2. Derive the camera matrix P from F
3. Estimate the 3D coordinates X from P & the 2D points x

Much of this section is covered in various parts of Hartley and Zisserman's "Multi-View Geometry" text (Hartley and Zisserman 2004) and can be referenced for additional information.

1. Derive F - The first step is to estimate the Fundamental Matrix F, from the point correspondences. From a set of n point matches $[x\ to\ x', y\ to\ y']$ we can use Equation (42) to develop linear equations of the following form,

$$x'xf_{11} + x'yf_{12} + x'f_{13} + y'xf_{21} + y'yf_{22} + y'f_{23} + xf_{31} + yf_{32} + f_{33} = 0 \quad \textbf{(181)}$$

$$[x'x \quad x'y \quad x' \quad y'x \quad y'y \quad y' \quad x \quad y \quad 1]f = 0 \quad \textbf{(182)}$$

$$Af = \begin{bmatrix} x_1'x_1 & x_1'y_1 & x'_1 & y'_1x_1 & y'_1y_1 & y'_1 & x_1 & y_1 & 1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_n'x_n & x_n'y_n & x'_n & y'_nx_n & y'_ny_n & y'_n & x_n & y_n & 1 \end{bmatrix} f = 0 \quad \textbf{(183)}$$

where the solution is the generator of the right null-space of A and F is a 3x3 matrix composed of the 9-element vector f. Specifics of the DLT implementation to solve for F are available in Chapter 12.

2. Deriving P from F - For a given 3-vector $e' = [e'_1 \quad e'_2 \quad e'_3]^T$ it is possible to define a skew symmetric matrix as follows (Hartley and Zisserman 2004) :

$$[e']_\times = \begin{bmatrix} 0 & -e'_3 & e'_2 \\ e'_3 & 0 & -e'_1 \\ -e'_2 & e'_1 & 0 \end{bmatrix} \tag{184}$$

Proof - The condition that $P'^T FP$ is skew symmetric requires $X^T P'^T FPX = 0$ for all X. Since $\mathrm{x'} = \mathrm{P'X}$ and $\mathrm{x} = \mathrm{PX}$, then $x'^T Fx = 0$, which defines the fundamental matrix. Now the following can be expressed,

$$l' = [e']_\times \left(P' P^\dagger\right) x = Fx \tag{185}$$

$$F = [e']_\times P' P^\dagger \tag{186}$$

$$H_\pi = P' P^\dagger \tag{187}$$

Additionally, since the fundamental matrix corresponds to a pair of camera matrices and due to the projection ambiguity, P can be chosen as,

$$P = [I|0] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \tag{188}$$

and its complement P' is,

$$P^{'} = \left[\left[e^{'} \right]_{\times} F | e' \right] \quad \textbf{(189)}$$

3. Derive X through P & x - Now, to develop an estimate of the 3D scene structure, we can utilize the camera matrix and linear triangulation methods. Since, $x = PX$ and $x' = P'X$, then we can combine them into the form $AX = 0$, which is linear in X.

$$x \times (PX) = 0 \quad \textbf{(190)}$$

$$x(p_3^T X) - (p_1^T X) = 0 \quad \textbf{(191)}$$

$$y(p_3^T X) - (p_2^T X) = 0 \quad \textbf{(192)}$$

$$x(p_2^T X) - y(p_1^T X) = 0 \quad \textbf{(193)}$$

where p_i^T are the rows of P and A which can be represented by,

$$A = \begin{bmatrix} xp_3^T & - & p_1^T \\ yp_3^T & - & p_2^T \\ x'p'^T_3 & - & p'^T_1 \\ y'p'^T_3 & - & p'^T_2 \end{bmatrix} \quad \textbf{(194)}$$

A similar approach is utilized when additional views are available; often with more robust results, since true triangulation can be utilized to improve the estimate. The solution for the 3-view projection matrix P, from the Trifocal Tensor T, utilizing the trifocal tensor notation, follows (Hartley and Zisserman 2004),

$$\left(l'^T [T_1 \quad T_2 \quad T_3] l'' \right) [l]_{\times} = 0 \quad \textbf{(195)}$$

$$\left(l'^{T}[T_i]l''\right)[l]_\times = 0 \tag{196}$$

$$l'^{T}\left(\sum_i x^i T_i\right)l'' = 0 \tag{197}$$

$$F_{21} = [e']_\times T_i e'' \tag{198}$$

$$F_{31} = [e'']_\times T_i{}^{T} e' \tag{199}$$

Again, due to projective ambiguity, the first camera can be chosen as

$$P = [I|0] \tag{200}$$

And, since F21 and F31 are known, the second and third cameras are

$$P' = [T_i e''|\mathrm{e'}] \tag{201}$$

$$P'' = \left[\left(e'' e''^{T} - I\right) T_i^{T} e'|\mathrm{e''}\right] \tag{202}$$

Although it will not be addressed here (see Section 11.3), it should be noted that the radial distortion coefficients can be incorporated into this solution space. A good example of this is addressed by Zhang (Z. Zhang 2000) and incorporates a linear estimation solution.

14.4 The Essential Matrix

Since we have knowledge of the WASP IOP, the Essential Matrix can also be utilized to estimate the 3D structure of a scene. The Essential matrix is defined below (Hartley and Zisserman 2004), where E embodies the relative rotation (R) and skew symmetric translation $[t]_\times$ between any two images of the same scene.

Essential Matrix

$$E = [t]_\times R \tag{203}$$

This is because the IOPs (K) Equation (139), can be applied to the camera projection matrix as,

Projection Matrix

$$P = K[R|t] \tag{204}$$

to obtain the normalized camera matrix .

Projection Matrix

$$K^{-1}P = [R|t] \tag{205}$$

Here, the Fundamental Matrix corresponding to two normalized cameras is commonly referred to as the Essential Matrix. Using the Camera Projection Equation, $x = PX$, and (205)

Normalized Coordinates

$$\hat{x} = K^{-1}x \tag{206}$$

then we arrive at the defining equations for the essential matrix .

Defining Equation

$$\hat{x}^T E\hat{x} = 0 \tag{207}$$

Additionally, it can be related to the Fundamental Matrix and IOPs via the following relationship,

Essential Matrix

$$E = \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{bmatrix} = K_1^T F K_2 \tag{208}$$

where K_1 and K_2 are the intrinsic calibration matrices of the two images. This relationship can be visualized in Figure 14-1 below.

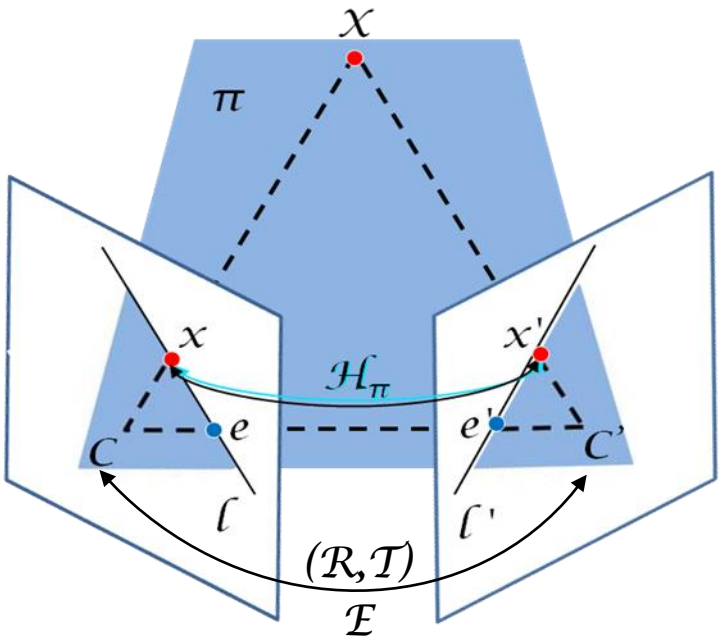


Figure 14-1 The Essential Matrix relates the two images using a simple 3D translation and rotation of the cameras.

Now we can implement the 3D structure recovery via a technique explained well by Yi Ma (Ma, et al. 2006). The Essential Matrix can be placed in vector form as follows,

Essential Matrix

$$E^9 = [e_{11} \quad e_{21} \quad e_{31} \quad e_{12} \quad e_{22} \quad e_{32} \quad e_{13} \quad e_{23} \quad e_{33}]^T \tag{209}$$

The Kronecker product (×) of two homogeneous coordinate vectors $\boldsymbol{x} = [x \quad y \quad z]^T$ and $\boldsymbol{x}' = [x' \quad y' \quad z']^T$, where $z = z' = 1$ is commonly utilized,

Kronecker Product

$$a = x(\times)x' \tag{210}$$

Coord$_H$ KronP

$$a = [xx' \quad xy' \quad xz' \quad yx' \quad yy' \quad yz' \quad zx' \quad zy' \quad zz']^T \tag{211}$$

Since the epipolar constraint of (207) is linear in the parameters of E, we can rewrite it as the inner product of a and E^9 as follows,

E KronProd

$$a^T E^9 = 0 \tag{212}$$

Now, we can define a matrix X_{nx9} of Kronecker Products, such that,

Matrix of KronProds

$$X_{nx9} = [a_1 \quad a_2 \quad \dots \quad a_n]^T \tag{213}$$

E & X

$$X_{nx9} E^9 = 0 \tag{214}$$

This linear equation can now be solved for the vector E^9 using an eight-point algorithm (Ma, Soatto, Kosecka, & Sastry, 2006) and the Eigenvector associated with smallest eigenvalue of $X^T X$ generating the values of E; which in turn can be "unstacked" into the 3x3 Essential Matrix. Now the relative pose (Rotation and Translation), embedded within E can be recovered and utilized with the image correspondences to retrieve the position of the point in 3D, by recovering their depths relative to each camera frame. Care must be taken to ensure structure results with a positive depth constraint and nonzero translation, since up to four possible results occur with calibrated reconstruction from E (Hartley and Zisserman 2004).

Now, the set of matching coordinates (x, x') can be utilized with the camera pose results to estimate structure λ, to within a uniform translation scale γ, using the following,

Rigid Body Pose Eq.

$$\lambda' \mathrm{x}' = \mathrm{R}\lambda\mathrm{x} + \gamma\mathrm{T} \tag{215}$$

Multiply by crossproduct

$$0 = \mathrm{x}' \times (\lambda' \mathrm{x}') = \mathrm{x}' \times (\mathrm{R}\lambda\mathrm{x} + \gamma\mathrm{T}) \tag{216}$$

Rearrange

$$\lambda x' \times Rx + \gamma x' \times T = 0 \tag{217}$$

Rearrange

$$M_i \bar{\lambda_i} = \begin{bmatrix} x_i' \times Rx_i & x_i' \times T \end{bmatrix} \begin{bmatrix} \lambda_i \\ \gamma \end{bmatrix} = 0 \tag{218}$$

Now we can solve the linear equation to estimate the depth component (λ_i) of each point correspondence.

14.5 Case Study – Creating Sparse Structure using Epipolar Geometry

In this case study, 12 images from RIT's WASP sensor were related into an image bundle to estimate the 3D terrain surrounding the VanLare Water Processing Plant. These images were processed through a SfM process developed primarily by Dr. Noah Snavely (Snavely, Bundler 2010), from the University of Washington, to produce a Sparse Point Cloud (SPC) of 3D points and relative orientation of the cameras (Snavely, Seitz and Szeliski, Photo tourism: Exploring photo collections in 3D 2006). This process was then commercialized by Microsoft into an online application called PhotoSynth (Microsoft Corporation 2010), which allows a user to upload imagery and view the resulting bundle of images and sparse structure (Figure 14-2).

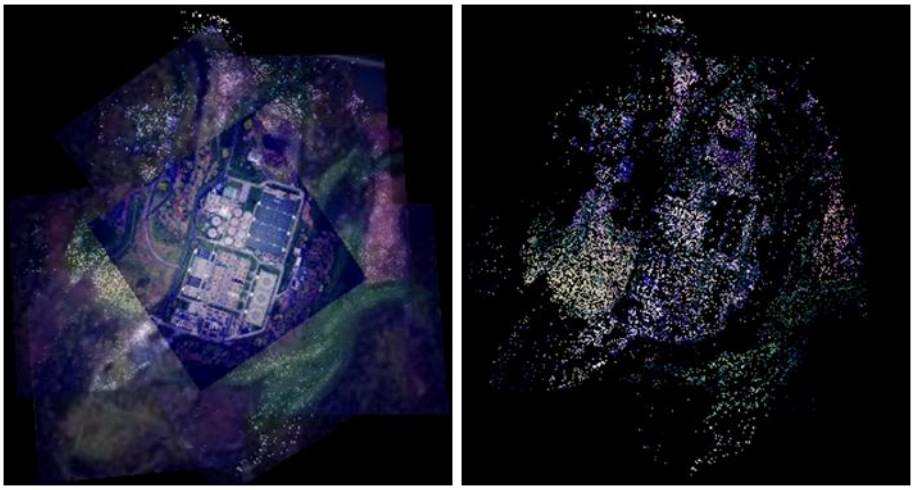


Figure 14-2 The graphics above show the results of Microsoft's PhotoSynth BA process.

These preliminary results show great promise in the ability to extract 3D information from 2D images, in a completely automated fashion, to produce relative relationships within a localized coordinate system. While these results are promising, there is currently no built in capability to export 3D structure or camera pose results in the freely available version. For this reason, the SBA software of Lourakis and Argyros (Lourakis & Argyros 2004), embedded within the Bundler code of Snavely (Snavely, Bundler 2010) was utilized to perform a similar recovery of structure and camera locations. Figure 14-3, shows the resulting point cloud and point cloud mesh overlaid onto GE terrain and models of the VanLare site.

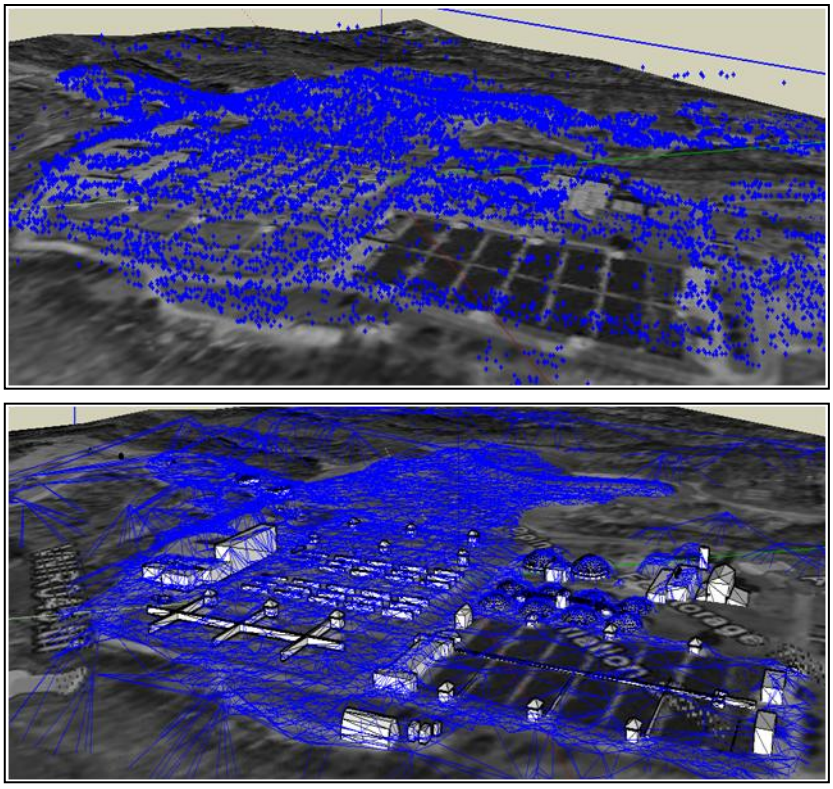


Figure 14-3 The SPC (top) and resulting mesh (bottom) from the Bundler SBA process (Snavely, Bundler 2010) using VNIR images from the WASP sensor.

15 APPENDIX E - Model Texture

In order to truly recover the 3rd Dimension from images registered to models, it is necessary to reapply the images onto the modeled environment. This can be done in a layered fashion, where the user has the ability to transfer between texture modalities as required or fused into three band images and viewed as entirely new products. Additionally, scrolling through a time-based series of images can have great advantage for temporal change analysis. Finally, products can often have unique characteristics, where the sum of the individual images is more useful than the individual components considered separately. This is evident in Figure 15-1, where the registered IR images of VanLare were stacked into a pseudo-color composite image, which highlights a newly constructed building composed of different building material than the rest of the plant.

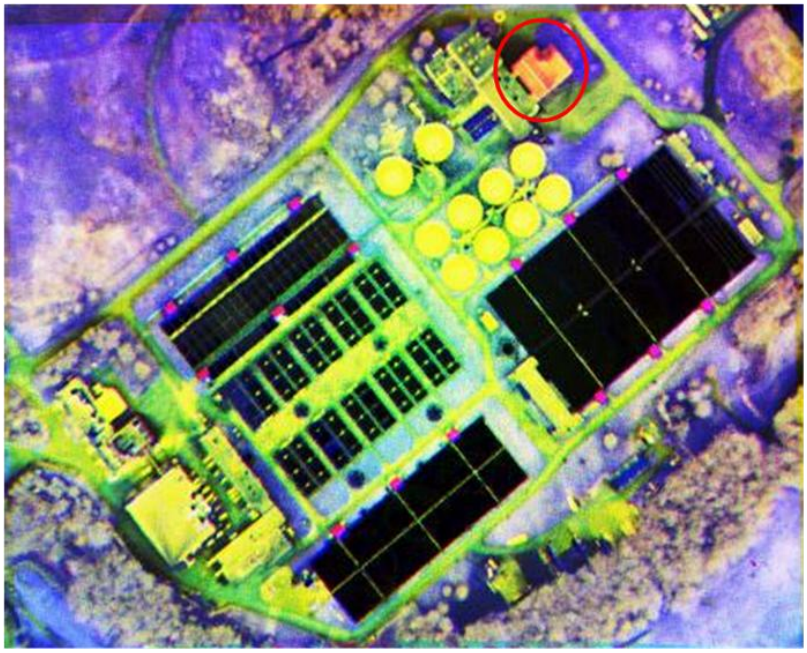


Figure 15-1 An illustrative example of IR image fusion in the form of a pseudo-color image stack. Circled in red is a new building that was constructed from different material (green metal) than the surrounding brick buildings with gravel roofs.

The following four techniques for texturing models with images offer distinct advantages and disadvantages and are largely dependent on the chosen application of interest.

15.1 Image Draping

Image draping is a useful technique to easily provide a rudimentary level of 3D texturing. Its application to near-nadir imaging was demonstrated in Figure 3-2, where it was utilized for comparison with similar overhead imagery. Below is an additional example of this technique from an IDL demo package (ITT Visual Information Solutions 2008).

Figure 15-2 By using a model (left) and related image (middle) it is possible to produce a realistic scene (right), as visualized using one of the demonstration tutorials within the IDL programming environment (ITT Visual Information Solutions 2008).

Here, an image is projected straight down, from nadir, onto the model surface. If done accurately, this draping/blanketing approach can provide a realistic model in non-urban areas. Unfortunately, modeled areas with near vertical features will display a stretched/smeared pixel appearance due to the way in which the texture is sampled and associated with the model. However facet surface normals could be used to test for this situation and texture exceptions could be incorporated to avoid undue smearing on the vertical edges of the model.

15.2 Facet Texturing and Model Unwrapping

This technique is a very efficient and realistic way to model an environment, thus owing to its longstanding popularity in the computer graphics industry. Unfortunately it is often a time consuming endeavor in the initial stages of "unwrapping" the texture and associating it properly to a model. Since we would like to automatically texture our models with numerous modalities and temporal updates, this can become an overly onerous option. Figure 15-3 shows the CIS building with multimodal textured facets.

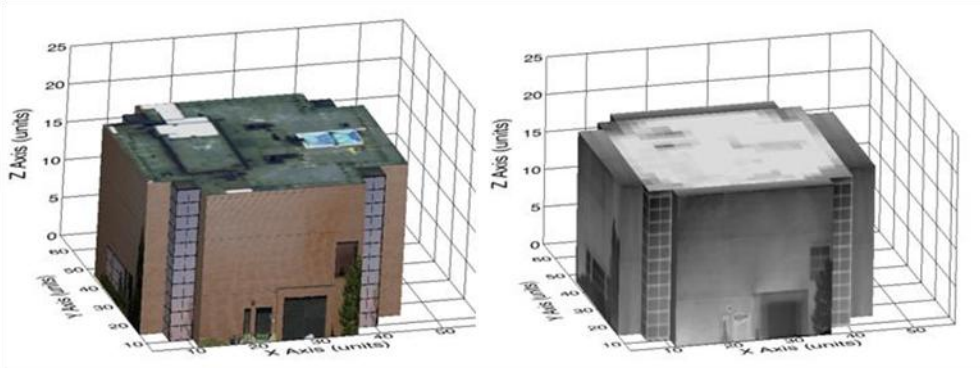


Figure 15-3 These multimodal models have been textured with image segments on each facet (visible-left & thermal-right).

In addition to the CIS model above, the VanLare site also utilizes this uv texture mapping approach to create visually realistic representations of the site. This model is embedded in a Collada format and placed in a Google Earth Keyhole Markup Language (KML) wrapper to associate the model with the world coordinate system as seen in Figure 15-4 below.

Figure 15-4 This realistic Pictometry model (Pictometry 2010) utilizes UV mapped oblique imagery to texture its facets and was then inserted into Google Earth (Google Earth 2010) using a KML description.

Finally, this uv texture mapping technique was utilized within the DIRSIG environment to build an accurate geometric and physical description of this same site for the purpose of 3D Multimodal registration. The basic process to accomplish this is again shown in Figure 15-5 for easy reference and was covered in detail earlier in Section 6.2.1.

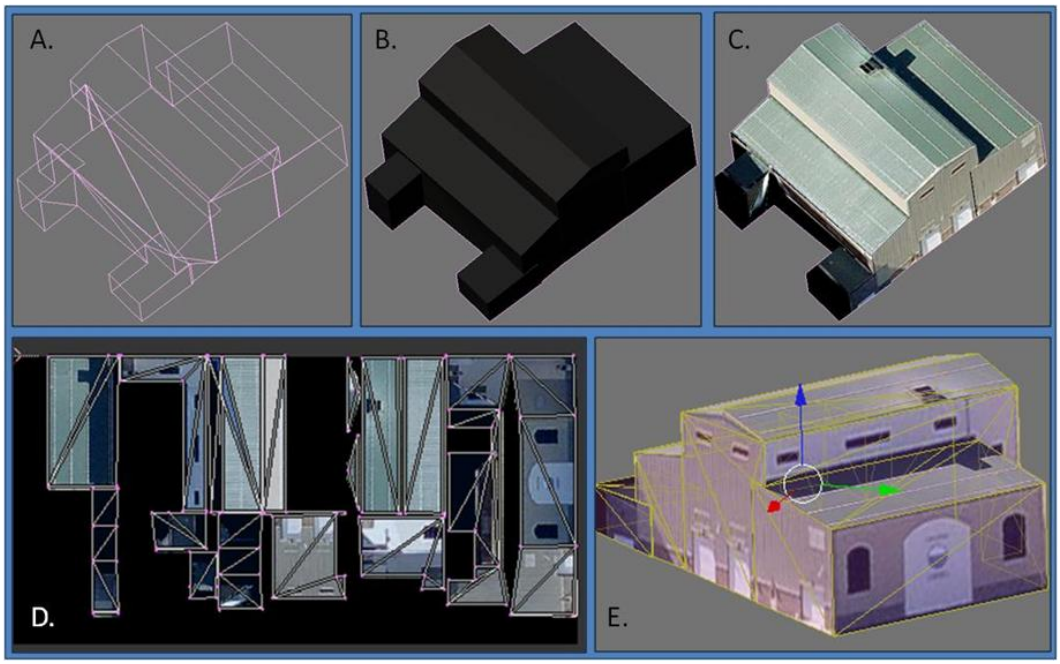


Figure 15-5 Illustrates the UV Texturing process: A) The wireframe model, B) The faceted model, C) The UV textured Model, D) The flattened (uwrapped) model with overlaying image texture, and E) The textured wireframe model.

15.3 Projective Texturing

The projective texture approach is one that the author highly recommends, since it can use the camera pose to project each pixel onto the scene accurately (Figure 15-6). Here the Google Earth's SketchUp (Google Earth 2010) application was utilized to perform the projective texturing. This technique could be implemented in such a way that the projected texture is only applied to surface facets if their normals are within a prescribed angular offset from the camera viewing direction (i.e. $\pm 45^{o}$). This would ensure that only minimal smearing would occur on the model facets that are parallel to the camera optical axis.

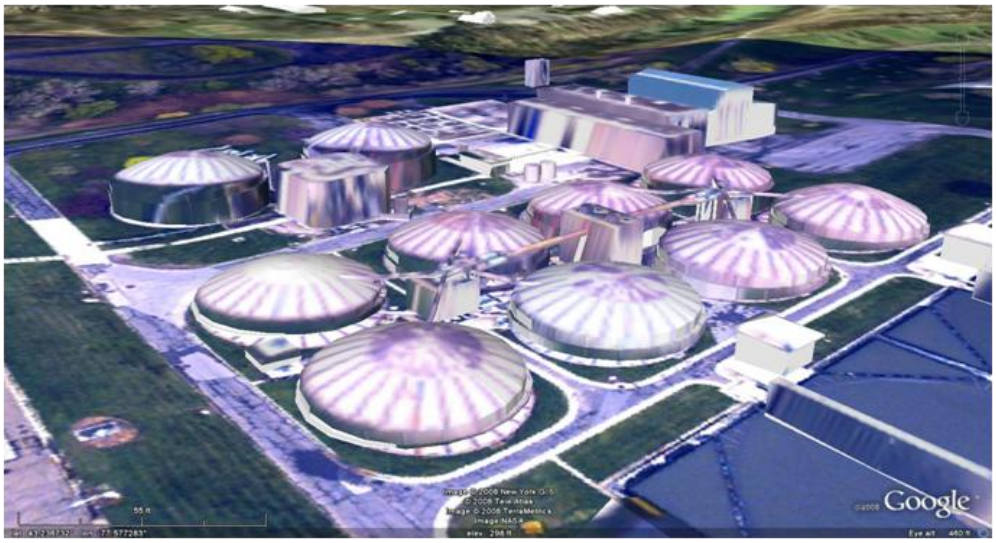


Figure 15-6 Here the same model has been textured using a projection tool in Sketchup (Google Sketchup 2009) and then imported into Google Earth (Google Earth 2010).

15.4 Volumetric Pixel (Voxel) Texturing

Since voxel techniques allow models to be developed as true volumetric datasets, they offer substantial benefits for atomic characterization of a scene. Additionally, this representation may allow the most accurate 3D reconstruction of a scene based off of SPCs and DPCs. The

ability to attribute each voxel with multimodal data, as seen in Figure 15-7, could also provide substantial advantages when trying to fuse these datasets. Unfortunately, the graphics industry has rallied around the faceted model approach, so few synergies of investment and research can be leveraged at this time.

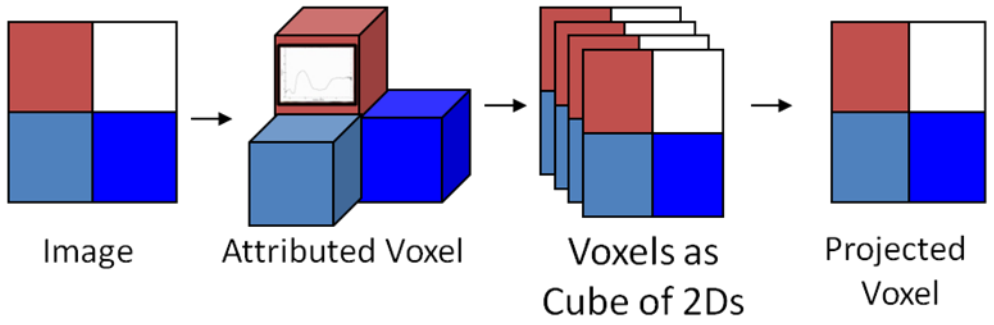


Figure 15-7 Volumetric Pixel (Voxel) approach to save data in volumetric space, but attribute as 2D facet.

16 APPENDIX F – DIRSIG Simulation Setup Primer

Any given simulation in DIRSIG will be encapsulated with a ".sim" description file. This file wraps various components of the simulation into a manageable whole and provides the essential links to various working directories. Embedded within the SIM file are calls to various DIRSIG modules describing the scene, atmospheric, sensor, platform, and data acquisition parameters (Figure 16-1).

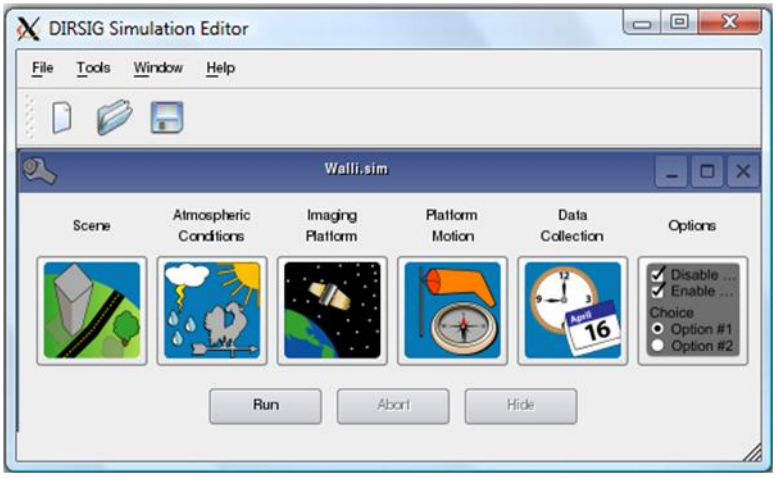


Figure 16-1 The DIRSIG Simulator Editor provides access to various components of the program.

16.1 DIRSIG's Scene File Setup

Pushing the Scene Icon in the Simulation Editor window will bring up another GUI which allows access into several other components within DIRSIG for defining the modeled scene characteristics. Pressing the "Geometry Tab" (Figure 16-2a) brings up another interface that allows definition of the Geographic Location, the Geometry List File (".odb") and linkages to the directory where the scene models are stored (Geometry Entity Directory).

DIRSIG's Scene Editor

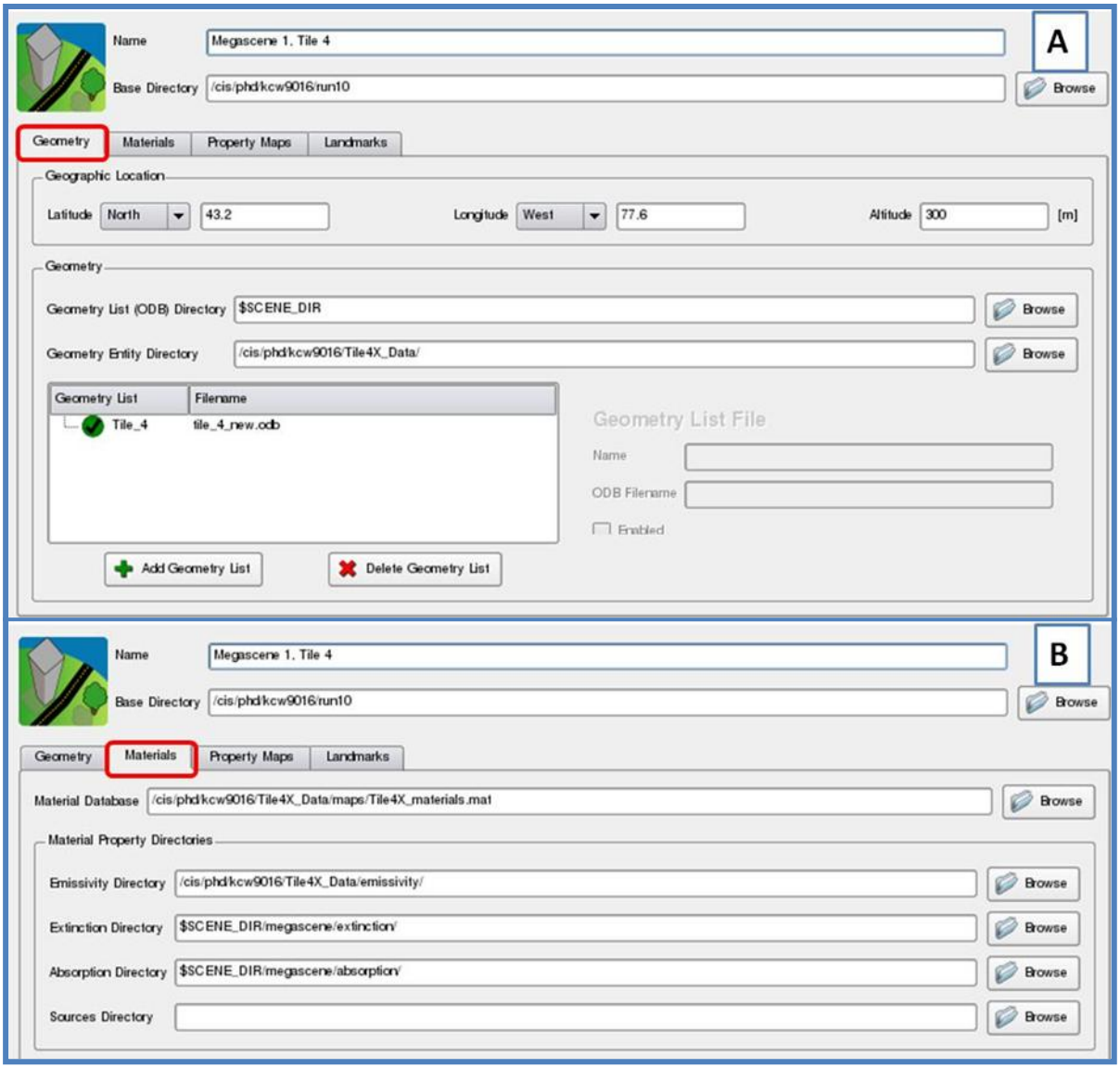


Figure 16-2 The Geometry tab (A), in the DIRSIG Scene editor, references the model geospatial and directory location, while the Material tab (B) links to the scene materials description file and emissivity file directory.

The "Materials Tab" (Figure 16-2b), similarly allows definition of a Materials File (".mat") and linkages to the appropriate emissivity, extinction, and absorption folders. The Material File describes the physical material characteristics used by DIRSIG to simulate scene content and is a critical component for multimodal registration (Section 6.4).

The "Property Maps Tab" shown below in Figure 16-3, is the workhorse for scene simulations.

DIRSIG's Property Map Editor

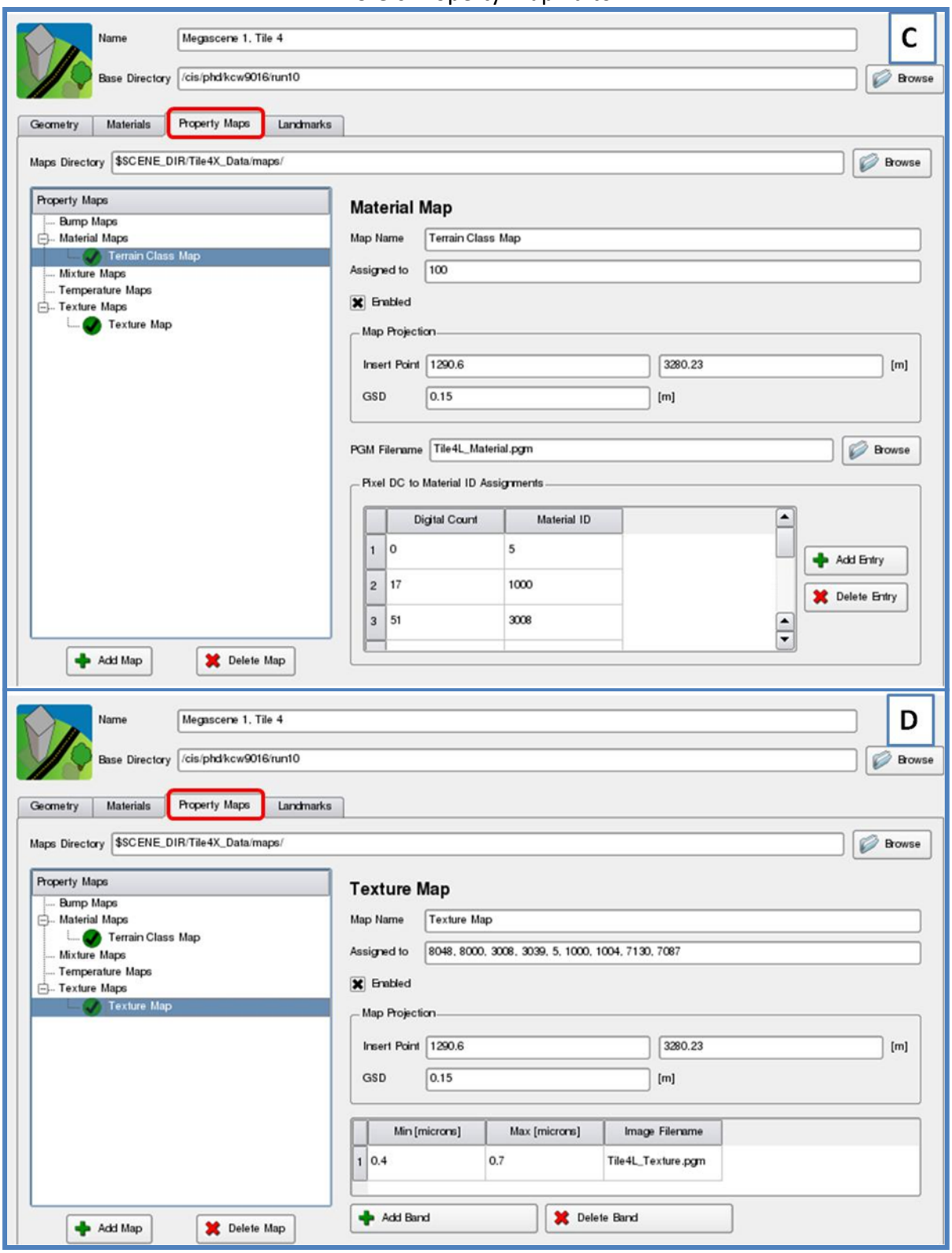


Figure 16-3 Within the Scene "Property Map" tab there are links (left panel) to the Material Map descriptions for the site (C) and Texture Maps (D). These "Property Maps" are tightly coupled within DIRSIG for physical scene description.

The window on the left pane allows the user to open additional interfaces into the “Material Maps” and associated “Texture Maps” sections of the application, which allow a user to define characteristics for the scene. The “Map Projection” section of both these interfaces allows for designation of localized offsets and the Geographic Sampling Distance (GSD) of both the Material and Texture Maps. In both graphics within Figure 16-3, the offset is referenced from the origin of MegaScene Tile-1 via the Insert Point text field in the Map Projection area.

Additionally, the “Material Map” interface (Figure 16-3c) allows the user to designate a specific model element in the “Assigned to” field. In the example above, ***ID = 100*** is used to identify a terrain “Material ID” within DIRISIG. It is important to remember that for every “Material/Texture Map” pair used in the simulation, a unique identifier must be generated to insure DIRSIG properly associates the maps and materials. The “Pixel DC to Material ID Assignments” section allows the user to assign a Look-Up-Table (LUT) that associates a discrete grayscale values to specific scene material characteristics via the DIRSIG Material File (“.mat”). These grayscale values are the Digital Count (DC) values of an image (“.pgm”) that has been segmented w.r.t. different scene materials and is designated in the “PGM Filename” field.

The associated “Texture Map” information is similarly accessed in the left window of the “Property Maps” tab by highlighting the appropriate Texture Map link (Figure 16-3d). The main difference between this and the previous interface is that the “Assigned to” field now contains the materials identified earlier in the “Material Map” assignments LUT and the user can designate the spectral bandpass and file linkage for the associated “Texture Map”.

16.2 DIRSIG's Sensor File Setup

By selecting the Imaging Platform icon from the DIRSIG simulation menu (top of Figure 16-4, highlight in red), the user can access several function that control the imaging sensor characteristics, as well as the platform and mount location and orientation. The mount interface can be accessed by selecting the mount link from the left pane of the System Components menu. This allows access to a handy tool for viewing the relative pointing of the platform and so it is possible to utilize the mount interface to insert the Pitch, Yaw, and Roll of the aircraft if the actual mount orientation is negligible or is incorporated into these values.

DIRSIG's Mount Editor

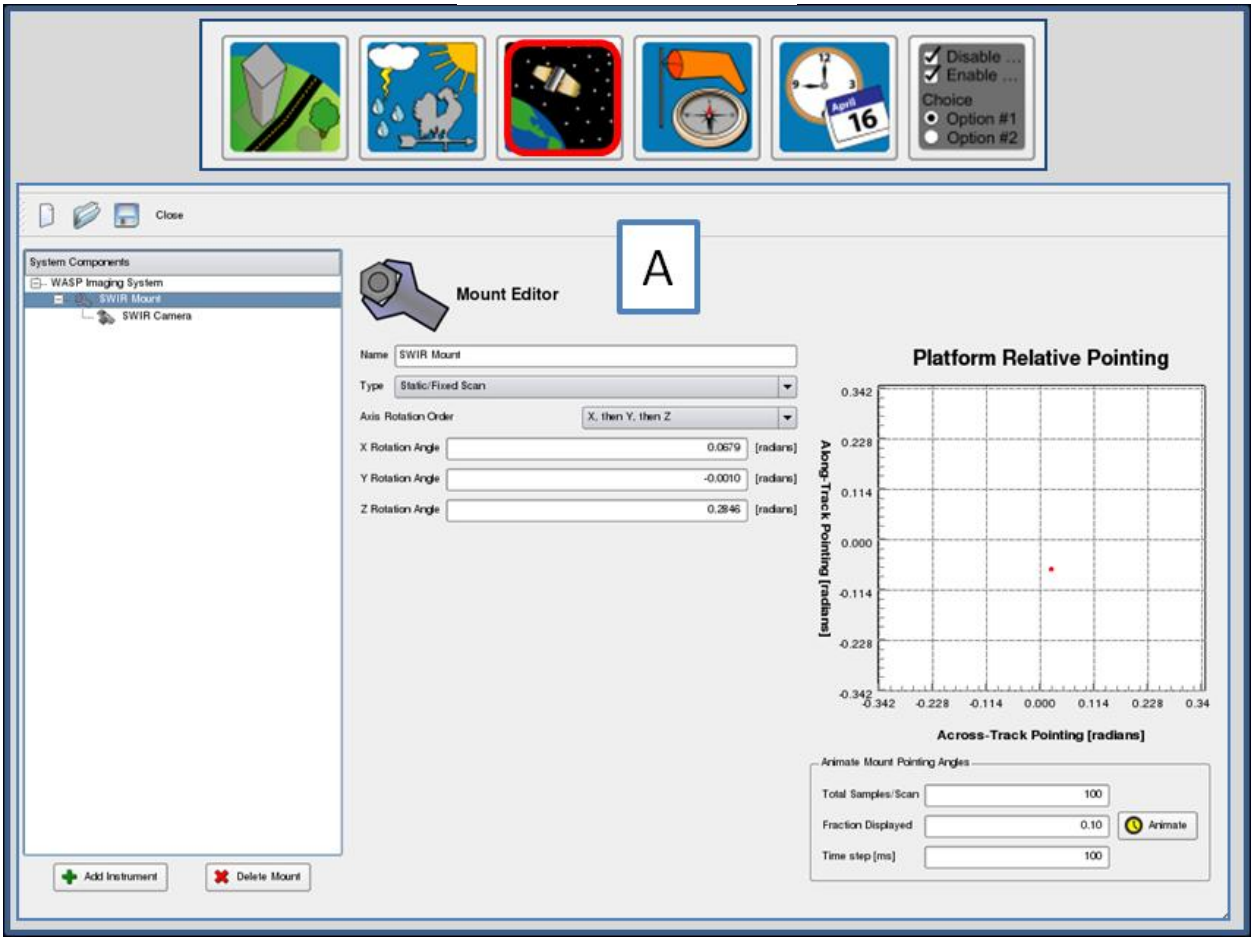


Figure 16-4 The Sensor Editor has links to a Mount Editor (A) and the Imaging Camera in the Left Panel. As seen here, the Mount interface was utilized to capture the sensor viewing angles which were retrieved from an Inertial Measurement Unit.

So, the sensor viewing angles can be injected into DIRSIG using either the sensor mount interface or the platform interface. In the figure above (Figure 16-4), the reader can see how the WASP Inertial Measurement Unit (IMU) data was converted to radians and inserted into the DIRSIG "Mount interface".

In order to access the "Instrument Editor", the user must highlight the last link on the left window pane in the "System Components" section (Figure 16-5). Here the user can insert additional mount offsets and orientations for multiple camera systems like WASP if desired. Additionally, the focal length of the camera can be edited within this interface and the user can access the camera's focal plane editor by highlighting the desired sensor in the "Focal Plane" section and then pressing the "Edit" button (Figure 16-5b). The "Focal Plane Editor" will describe the sensor's "Array Dimensions", "Pixel Pitch" and the "Spectral Response/Range".

DIRSIG's Sensor Editor

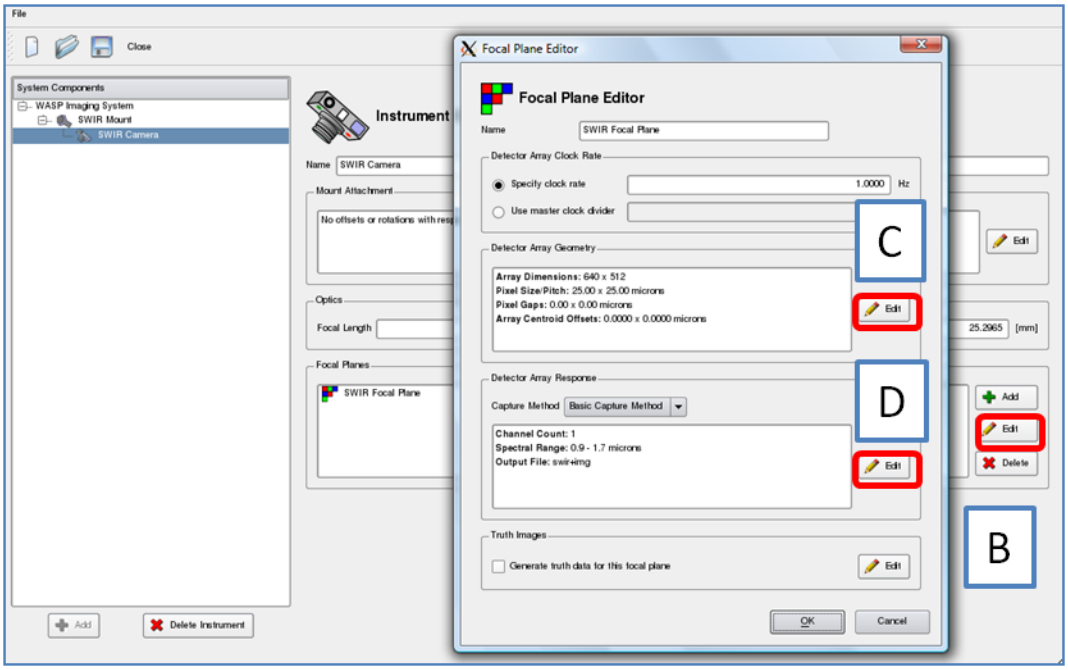


Figure 16-5 Within the Camera Instrument editor, there is an "edit" button for the Focal Plane (B). Pressing this button will bring up the Focal Plan Edit menu with additional buttons for editing the Detector Array (C) and the Response Curve (D).

These parameters can in-turn be accessed and changed by pressing the corresponding “Edit” button in the “Detector Array Geometry” and “Detector Array Response” panes Figure 16-6).

DIRSIG’s Focal Plane Editor

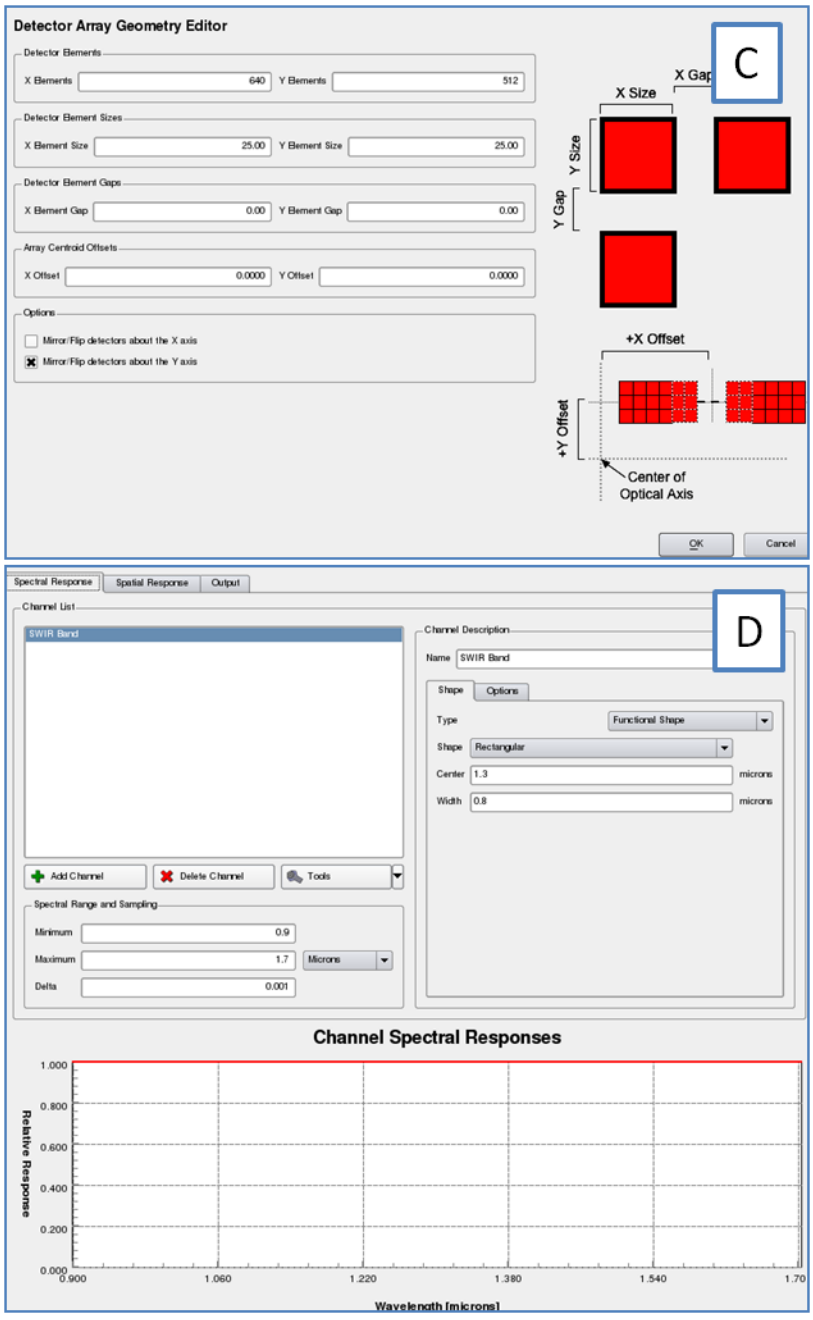


Figure 16-6 The Focal Plane editor buttons bring up the Detector Array editor (C) and Detector Spectral Response editor (D) windows, which allow a great deal of flexibility in defining the sensor specific design characteristics.

16.3 DIRSIG's Platform File Setup

The Platform File editor provides a convenient description of the vehicle (satellite, airplane or terrestrial) that transports the imaging sensor. Through this interface, it is possible to define the coordinates and orientation of the imaging platform and even the rotation order (Figure 16-7).

DIRSIG's Platform File Editor

Figure 16-7 The Platform Editor allows for the designation of geospatial position information, such as Latitude, Longitude, Altitude and the orientation information of the sensors External Orientation Parameters, such as Pitch, Yaw & Roll.

There are a few things that should be noted when entering the WASP platform information into DIRSIG. First, the DIRSIG MegaScene Tiles were all based off of a local coordinate system, where the lower left corner of Tile-1 is regarded as the origin. Tile-4 has an origin that is located at UTM Coordinates Longitude = 289,826 [m] and Latitude = 4,789,892 [m] (Zone 18T), with local MegaScene coordinates of Longitude = 1291 [m] and Latitude = 2330 [m]. In order to

compare the DIRSIG simulation results to real imagery, it is necessary to convert between the global UTM coordinates of the WASP flight data and this local coordinate system (Figure 16-8).

Converting WASP Flight Data into DIRSIG Platform Parameters

	Longitude [m]	Latitude [m]	Flying Alt [m] (above Geoid)	Geoid Delta [m]	focal length [mm]
WASP VNIR045 UTM	290,707.26	4,790,203.71	879.42	35.85	
Tile-4 Origin UTM	289,826.0	4,789,892.0			
Local Offset	881.26	311.71			
Tile-4 Offset	1,290.60	2,329.80			
DIRSIG Input	**2,171.86**	**2,641.51**			
VNIR045	**Degrees**	**Radians**	879.42	**915.27**	**55mm**
Omega	4.49152	0.078391812			
Phi	0.10469	0.001827185			
Kappa	17.19189	0.300055085			
SWIR078	**Degrees**	**Radians**	879.596	**915.446**	**25.2965**
Omega	3.88848	0.067866779			
Phi	-0.055	-0.000959931			
Kappa	16.30377	0.284554467			

Figure 16-8 In order to properly inject the WASP GPS/IMU data into DIRSIG it is essential to convert for any local coordinate translations, sensor angles and Geoid offsets. For the VanLare site, this offset accounts for 36 [m] higher flying altitude.

Secondly, as mentioned earlier, the sensor's view angles (EOPs) were captured in DIRSIG using the Sensor Mount Interface after a Degrees-to-Radians conversion of the flight data was performed. For this reason, the orientation parameters are included as null offsets in the platform file editor interface.

Finally, due to the localized error from the Geoid model height, it may be necessary to account for this offset within DIRSIG. Since the WASP GPS sensor delivers flying height above the Geoid in its flight data, localized variation in the terrain should be incorporated if available to ensure

the simulated imagery scale is as close to "truth" as possible. This becomes especially important when deriving the 3D scene geometry from hi-resolution WASP VNIR images. The local DIRSIG coordinates for the original five MegaScene Tiles is shown below in Figure 16-9.

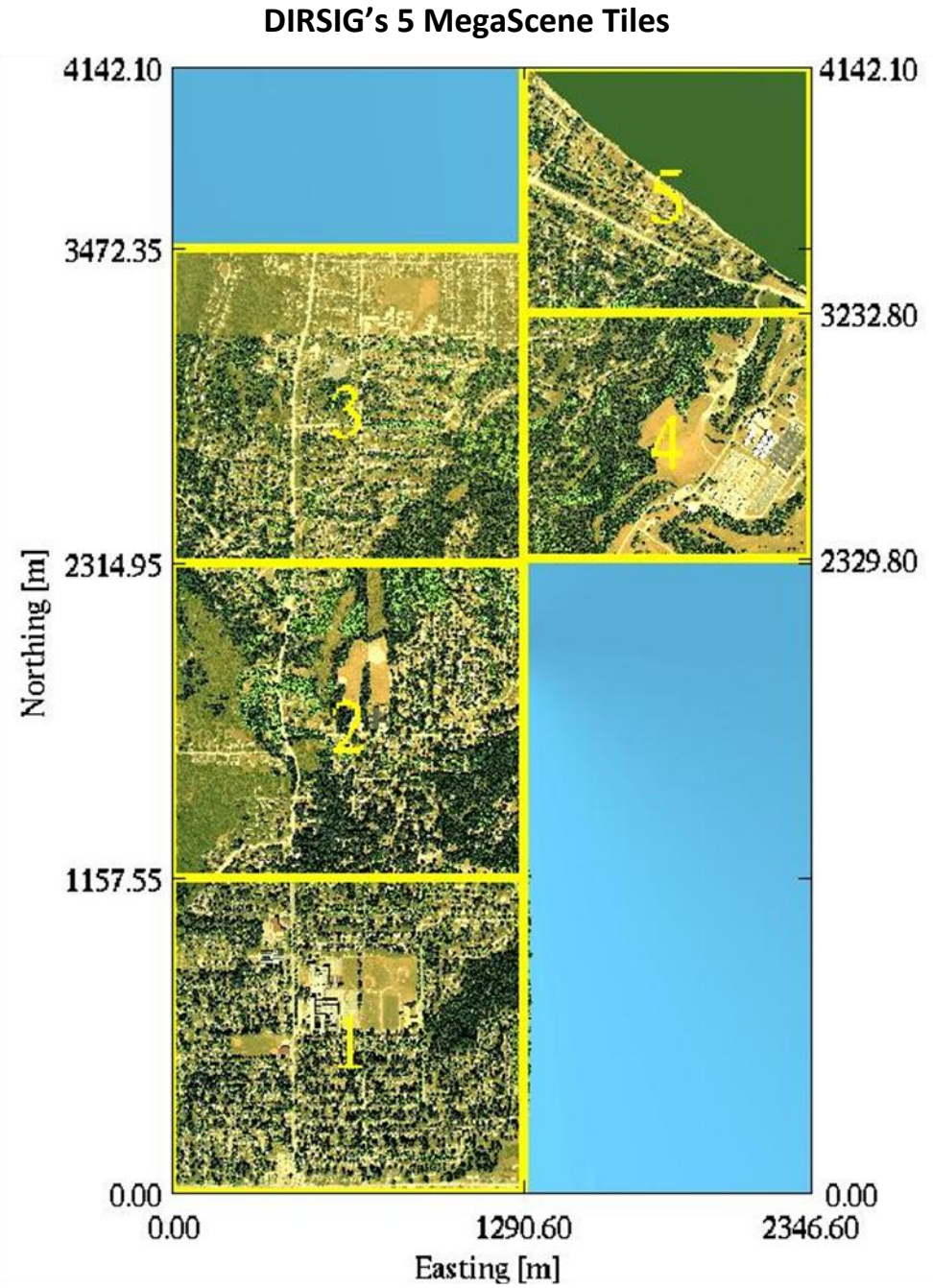


Figure 16-9 DIRSIG's 5 MegaScene Tiles (courtesy Mike Presnar) cover a swath of Northern Rochester and include a variety of environmental settings, including residential, agricultural, industrial, and lake frontage. The VanLare test site is in Tile-4.

16.4 DIRSIG's Atmospheric Conditions Setup

The Atmospheric Conditions editor within DIRSIG has two tabs which allow input of the weather conditions at the time of simulation and specifics regarding the radiation transport of the photons through the atmosphere. The radiation transport tab provides links to the MODTRAN Tape-5 file and the atmospheric database file that is generated at the beginning of a DIRSIG simulation, providing essential atmospheric LUT parameters. This user interface is visible below in Figure 16-10.

DIRSIG's Atmospheric Conditions Editor

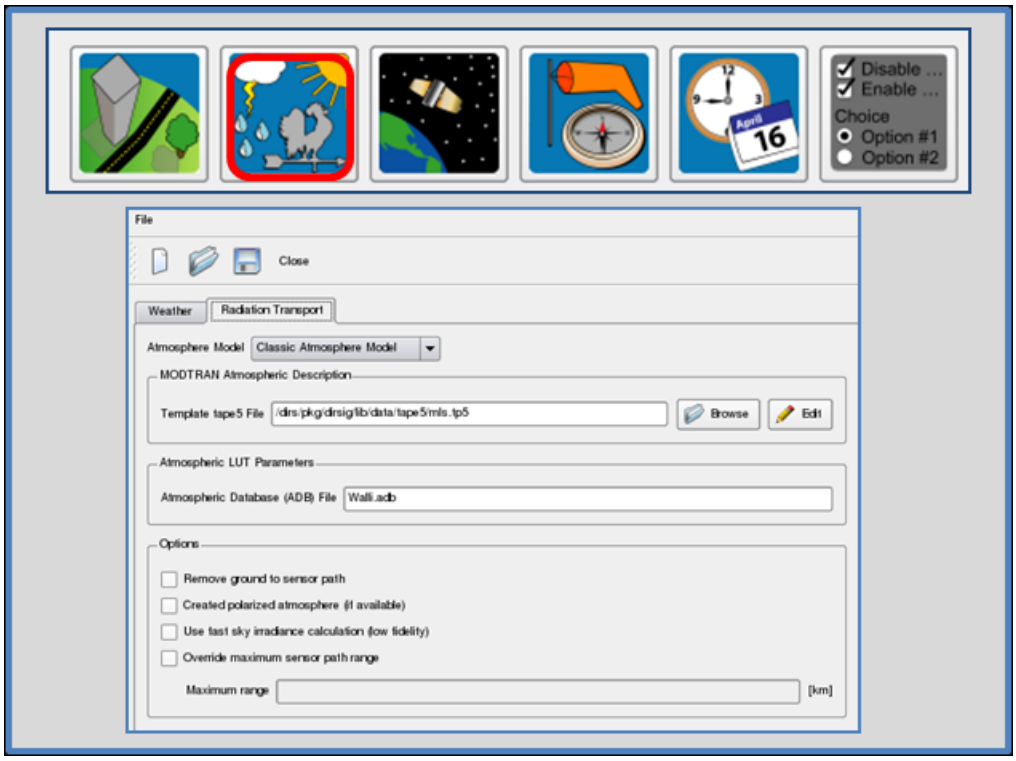


Figure 16-10 The Atmospheric Conditions Editor allow for designation of the Weather conditions at the time of the collection and the designation of Radiation Transport parameters via MODTRAN Tape-5 files.

16.5 DIRSIG's Data Collection Setup

The DIRSIG Data Collection GUI can be utilized to specify single frame or multi-frame (video) output. Additionally, the user can specify an instantaneous image capture of the modeled

scene or an integrated exposure over time. Finally, the time of image capture can be specified. This is a very important feature for simulating real imagery collections for registration, since it makes it possible to estimate the scene shadowing correctly w.r.t. the solar zenith angle. This DIRSIG user interface is visible below in Figure 16-11.

DIRSIG's Data Collection Editor

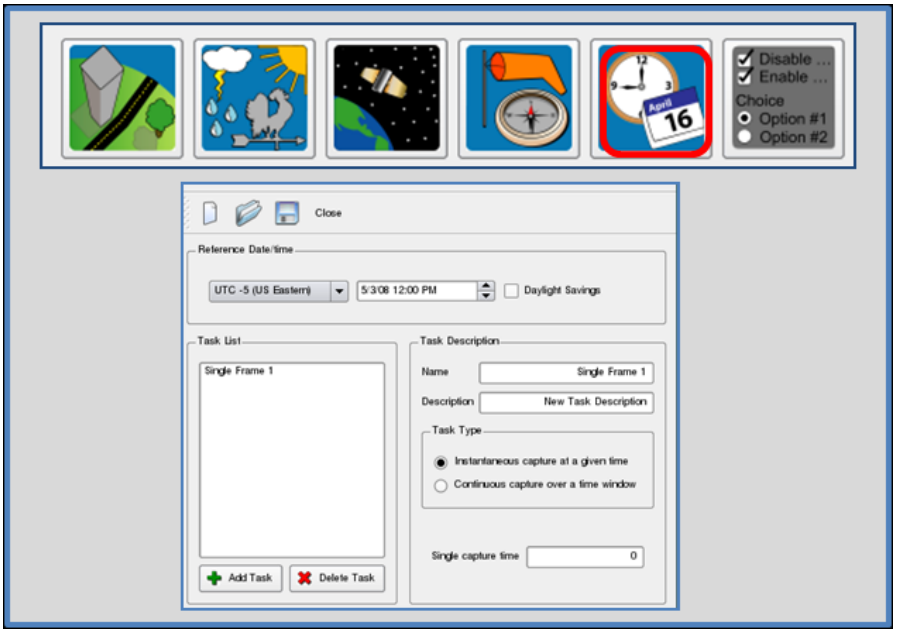


Figure 16-11 The Data Collection Editor allows the user to designate the day and time of collection; this is essential for properly casting shadows onto the scene from the correct solar position.

17 APPENDIX G – MATLAB Software Flowchart and Index

The included MATLAB (The Mathworks, Inc. 2010) programs contain many helpful tools that can allow the user to integrate various aspects of this research into their daily registration workflow. These programs are available upon request to either the author or members of the dissertation committee.

17.1 Image Registration

The following flowchart shows the hierarchy of program execution and basic components for image registration using the provided MATLAB tools and case study data.

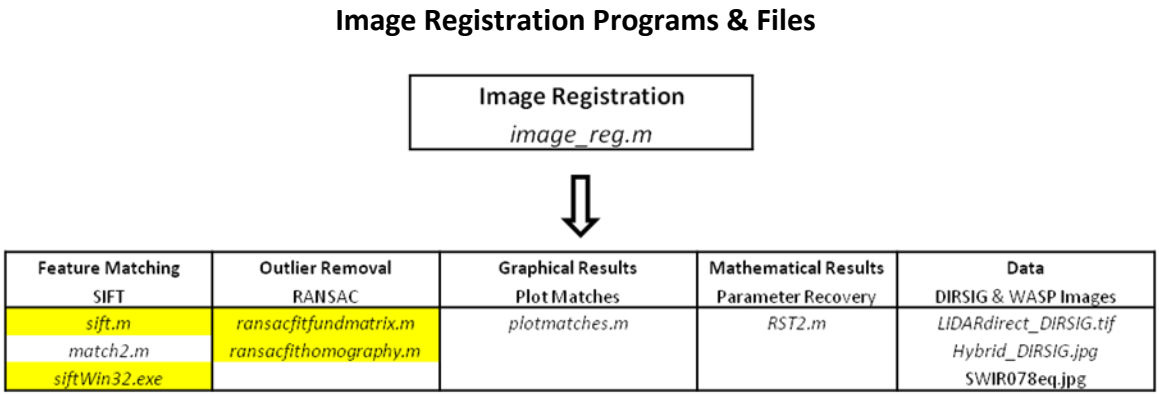


Figure 17-1 This flowchart provides a snapshot of the tools provided for image registration and the related file structure (programs highlighted in yellow were not written by the author).

17.2 Sparse Point Cloud Generation

The following flowchart shows the hierarchy of program execution and basic components for Sparse Point Cloud generation and depth-map recovery using the provided MATLAB tools and case study data.

Sparse Point Cloud Programs & Files

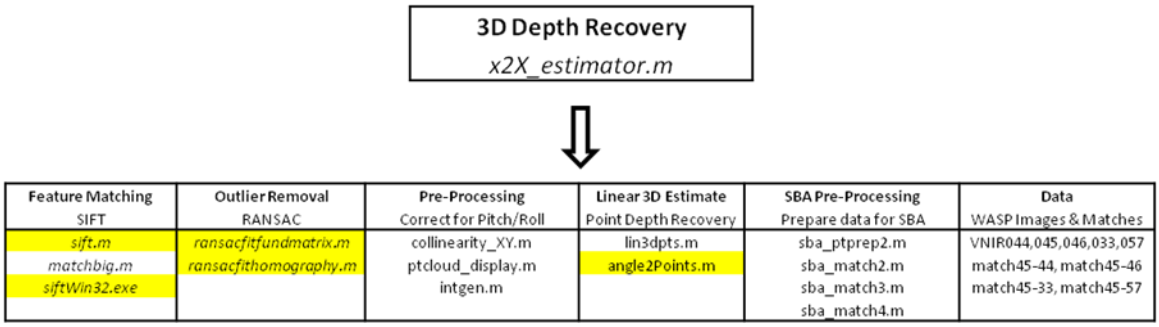


Figure 17-2 This flowchart provides a snapshot of the tools provided for SPC Generation and the related file structure (programs highlighted in yellow were not written by the author).

17.3 Model Registration & Pose Estimation

The following flowchart shows the hierarchy of program execution and basic components for a *Pose Estimation* of a 3D model as viewed from an image using the provided MATLAB tools and case study data.

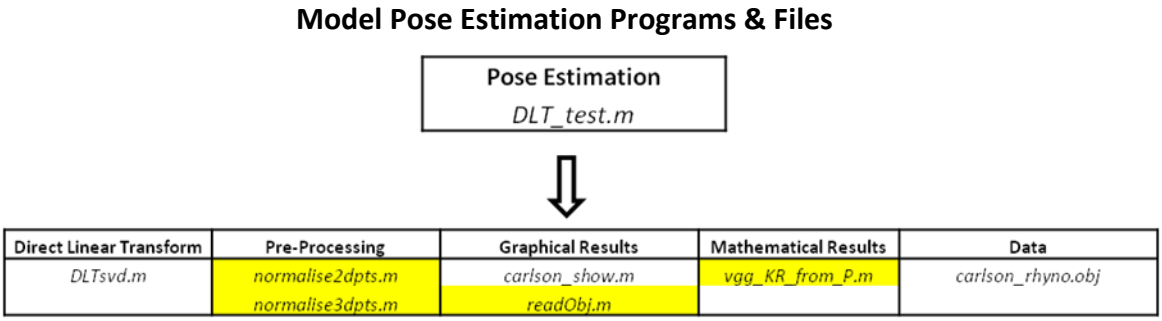


Figure 17-3 This flowchart provides a snapshot of the tools provided for Pose Estimation and the related file structure (programs highlighted in yellow were not written by the author).

Additionally, a similar graphic is available for 3D Model Registration using the authors 3D Conformal Transform (rigid body) and case study data used to relate the AANEE model to the World Coordinate System using Google Earth (Google Earth 2010).

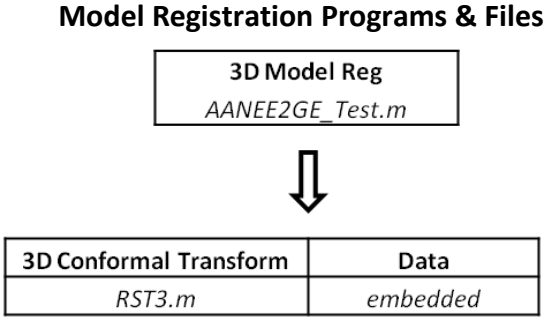


Figure 17-4 This flowchart provides a snapshot of the tools provided for Model Registration and the related file structure.

17.4 LIDAR Data Processing

The following flowchart shows the hierarchy of program execution and basic components for processing of LIDAR data to extract Regions of Interest (ROI) from a ".las" formatted file. Once accomplished, this 3D Dense Point Cloud ROI can be used as the basis for a facetized model using the provided MATLAB tools and case study data.

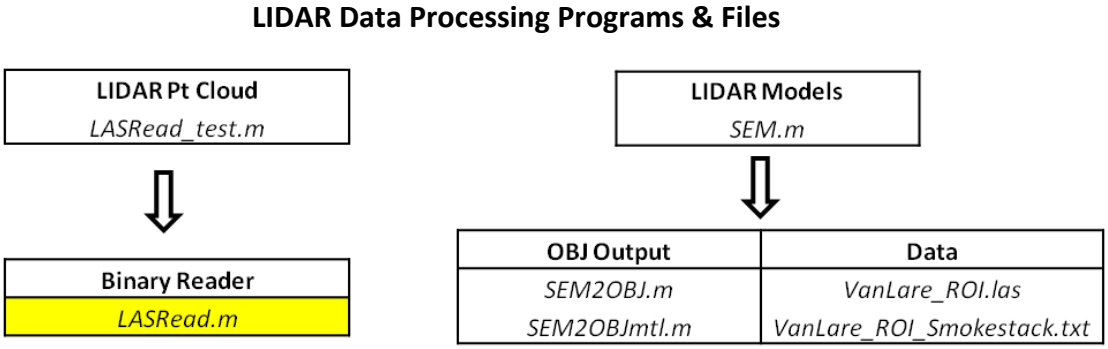


Figure 17-5 This flowchart provides a snapshot of the tools provided for LIDAR Processing and the related file structure (programs highlighted in yellow were not written by the author).